Commodore INFO

IDEEËN-WEDSTRIJD

Win een C-128 of Seiko polsterminal

ONAFHANKELIJK BLAD VOOR COMMODORE GEBRUIKERS PRIJS ƒ 6,25/Bfr. 125 JAARGANG 2, No. 6, 1985

ZOMER-NUMMER



CES software

Machinetaal

CP/M op de 64

Schermwerk

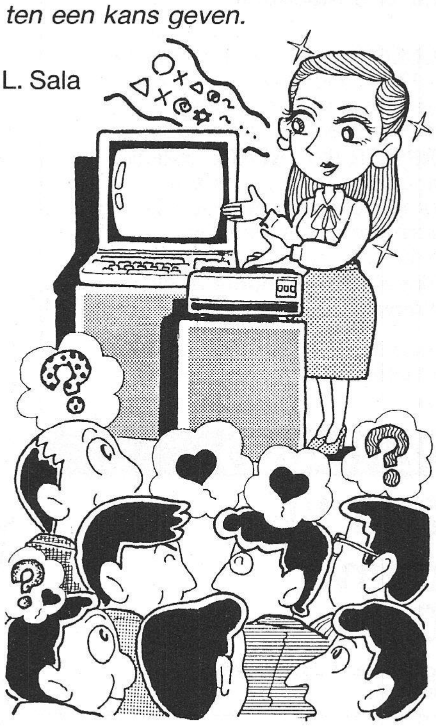
Extra veel listings

Commodore Info 14 september Computerbeurs Den Haag

Redactioneel

Het grote Commodore-Offensief gaat eindelijk van start. We hebben er allemaal op zitten wachten, op de machines die de echte sprong vooruit zouden betekenen na de toch matige indruk die het 264, 364, Plus/4 en C-16 experiment achterliet. En we waren het wachten eigenlijk wat moe!
Maar goed, de machines zijn er en de 128 komt hopelijk volgende maand in de winkel. De knaller voor de kenner, de supermicro AMIGA werd op 23 juli in New York voor het eerst getoond. We waren erbij en u hoort in het volgend nummer meer over deze doorbraak naar professionele creativiteit. Indrukwekkend, om je vingers bij af te likken, om stil van te worden.
Ook bij Commodore-Info zelf staat de ontwikkeling niet stil. We zijn erg gegroeid dit laatste jaar en dus betrekken we in Amsterdam een nieuwe kantoor (Den Texstraat 5a), waar Joost Taverne als uitgever nu verantwoordelijk is voor de commercie en administratie. De bedoeling hiervan is dat ikzelf dan wat meer tijd kan besteden aan het schrijven van dit en andere bladen en nieuwe activiteiten kan opzetten.
Ook komt er weer een Commodore-Info beurs (op 14 September in Den Haag) en gaan we middels een nieuwe Ideeën-wedstrijd uw creatieve talenten een kans geven.

L. Sala



IN DIT NUMMER:

COMMODORE-INFO
Jaargang 2, no. 6
Dit blad wordt uitgegeven door:
SAC, PB 112, 1260 AC te Blaricum, NL.
tel. 02152-65695.
Commodore-Info is een onafhankelijk blad en verschijnt tenminste 10 x per jaar.
Abonnement f 55,– per jaar op giro 1585491 tnv SAC Blaricum.
Abonnementenadministratie:
Den Texstraat 5A, 1017 XW Amsterdam
Tel. 020 - 273198

Voor België: B.B.Lambert,
(f 55,– of Bfr 1100)
banknr. 310050602562
tnv SAC Blaricum.

Distributie: Betapress Gilze
voor Belgie: AMP Brussel

ISSN 0169 -3085

Druk: Drukkerij Verweij B.V. - Mijdrecht



LUC SALA'S DATAKOLOM

Is de computergebruiker een slaaf van zijn beeldscherm, zelfs al is dat plat? Blijven zijn vingers de belangrijkste transmissiefactor via toetsenbord of muis?
Ik geloof de profeten van de microwereld eigenlijk maar half, als ze weer zo'n schitterend verhaal ophangen over mensvriendelijke bediening en dan met ikonen, user interface en touchscreens aan komen dragen. Prima ideeën allemaal, maar heeft u ook niet het gevoel, dat dit maar een kant van de mogelijke computer-medaille betreft. Want de mens is meer dan ogen en vingers, we kunnen ook horen, voelen, ruiken om maar een paar zintuigen te noemen. En dan heb ik het nog helemaal niet over telepatisch contact met de chips, dat is weer een andere dimensie.

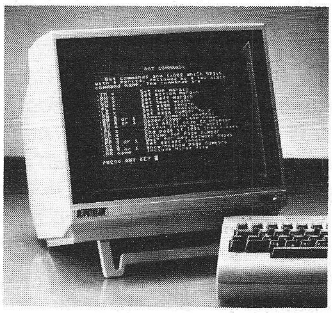
Laten we eens een compleet andere computer voor de geest halen, een waarbij ogen en vingers geen rol spelen. Een leuk voorbeeld zou de AUDIOCOMPUTER kunnen zijn. Dat is een apparaat, waarbij geluid wordt gebruikt om met de computer te communiceren in twee richtingen. En we gaan dan een stapje verder dan de stemgestuurde probeerseltjes op hedendaagse micro's, dat zijn in wezen erg simpele alternatieven voor besturingscommando's. Want in plaats van intypen COPY of DIR kun je dat dan inspreken. Leuk, maar nog steeds gebonden aan het huidige computerconcept. Kijk eens een stapje verder, naar een systeem waarbij bv. audiosignalen als muziek of tekst interactief op iedere gewenste plaats af te spelen zouden zijn. Het daarmee zelf programmeren van de afspeelvolgorde van een bepaalde LP of CD is maar een simpel voorbeeld en trouwens ook één, dat we binnenkort ook wel op standaard audioapparatuur kunnen verwachten.

Een Audiocomputer kan net zo interactief zijn als een videocomputer.

Er bestaat natuurlijk nog helemaal geen Audiocomputer, iedereen is nog veel te druk met de Videoversie bezig. Toch lijkt me, dat bijvoorbeeld zo'n nieuwe Sony Walkman waarop met TWEE cassettes tegelijkertijd kan worden gewerkt, of met een beetje flexibele CD CompactDisk al heel wat te bereiken zou zijn. Bijvoorbeeld een talencursus, met op de ene kant de lesstof en aan de andere kant de oefeningen. De bestuurshardware en software, en daarmee dus het hart van de Audiocomputer, zou steeds kunnen zorgen voor de optimale afstemming tussen die twee. En heel wat lesstof is toch overdraagbaar via geluid en spraak. En wat dacht u van rondleidingen in bv. musea, waarbij met simpele herkenning van bv. stop en go het tempo valt aan te passen. Of conditietraining met mondelinge begeleiding, ochtendgymnastiek, in wezen alle muzikale interactie, waarbij middels wat chips de tempobepaling kan worden beïnvloed. Een menugestuurd spelletje op weg naar school of werk, een tekstadventure met een simpele audiojoystick, die reageert op hoog/laag/links/rechts, het is echt niet zo moeilijk te realiseren. Gaan we nog een stapje verder en nemen we aan dat stemherkenning echt toepasbaar wordt, dat groeit de toepassing van de audiocomputer navenant. Zoals bij het hele telefoonverkeer, het afhandelen en vastleggen daarvan is toch typisch een audiotaak, waarbij een toetsenbord of beeldscherm in wezen overbodig zijn. "Bel Jansen" zou toch voldoende moeten zijn voor een beetje chipskolonie in de vorm van onze imaginaire Audiocomputer. En wat te denken van beveiligingstaken, voorlopig schrikt een autoinbreker waarschijnlijk nog wel van een direct tot hem gerichte tekst. Kinderoppas met interactieve verhaaltjes voor het slapen gaan, we kunnen dit ook nog doortrekken tot praktische maar ook wat beperkte toepassingen als een automatische gesprekspartner in het café, de oplossing om van het gezeur in de echtelijke en andere spondes af te zijn. Maar ja, dan zijn we in de situatie waarin we niet meer weten, wie nu antwoordt, de mens of de machine en dat is ooit als criterium voor "denkende" machines gebruikt.

nieuws

Slipstream Hi-res monitor



Van Dunnet is er een nieuwe hi-resolution monitor op de markt, die speciaal ontworpen is voor gebruik met huiscomputers: de **Slipstream Monitor** met een amberkleurige beeldbuis (DF622) of met een groene beeldbuis (DF 625) voor *f* 449,-.
De monitor heeft een bandbreedte van 20 MegaHerz en een resolutie van 850 lijnen in het midden van het beeld. De maximale display is 720 x 312 pixels, dus geschikt voor 80 kolms en C-128. Door de uitvoering van de beeldbuis met P 42 fosfor (gemiddelede nalichttijd) is er sprake een goede beeldstabiliteit voor grafische en teksttoepassingen.

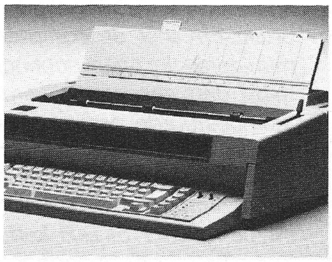
Koala Light Pen

Dit van de Koalapad bekende bedrijf brengt nu (via importeur Dunnet) ook een lichtpen. Niet goedkoop voor ruim 245 gulden, maar over kwaliteit gesproken, dit zijn de meesters van het grafische werk. En er is het programma PENPAINTER bij, zodat de creatieve geesten direct aan het werk kunnen en wel pixel voor pixel, want zo gedetailleerd valt hiermee te werken.

Leer Bridge

Al weer een tijdje geleden kwam bij ons op de redactie iemand op bezoek met een nieuw Bridge programma. Het zag er fraai uit en hij was op zoek naar een manier om het commercieel te exploiteren. Dat is ondertussen gelukt, oa. bij MCN en het softwarehuis BRIDGESOFT heeft het pakket nu dus in de winkels liggen. We hebben wat kunnen spelen met LEER BRIDGE en het is ook grafisch een goed uitgevoerd programma, dat zich bovendien houdt aan de in ons land gebruikelijke ACOL biedconventies, en dus aansluit bij een TV-cursus over Bridge van de Avro dit najaar.

IBM printers



U weet waarschijnlijk wel, dat in ons land ook de wereldwijd befaamde IBM bolletjes-schrijfmachines werden gemaakt. Er zijn in het verleden wel pogingen gedaan, om die machines te gebruiken als computerprinter, maar zonder veel succes. Nu de opvolgers van de bolletjesmachines er zijn, heeft IBM wel rekening gehouden met de printerfunctie. Let op, IBM zou wel eens een belangrijke rol kunnen gaan spelen op printergebied en dan met een "nederlands" produkt.

De C-128 in Keulen

Tijdens de in juni jl. gehouden Keulse C'85 Internationale Computershow was het behoorlijk vol. Commodore produkten van vele kanten trokken zoals gebruikelijk de meeste belangstelling, vooral in het weekend. Commodore heeft daar de nieuwe C-128 aan het Duitse publiek gepresenteerd na de "halve" introductie tijdens de Hannover Messe in april van dit jaar. De machine was er toen al wel maar werkte alleen nog maar in de 64 mode.



Het is overigens nog steeds niet helemaal duidelijk wanneer de C-128 in ons land leverbaar zal zijn. Er lijken enkele problemen met de CP/M ROM te bestaan, waardoor het waarschijnlijk iets later wordt. We hopen nu, dat de machine in september 1985 in de winkels ligt. de aanvankelijk geplande datum.

Top 20

Niet alleen in de muziekhitlijsten heeft het Engelstalige repertoire al jarenlang de overhand, ook op de software charts domineren de Angelsaksische (en soms Japanse) pakketten. Dat is niet alleen het geval in Nederland, ook de Duitse C-64 Top 20 van vorige maand bevat maar één duitstalig programma. Uit het hiernavolgend overzichtje blijkt de voorkeur van de Duitse C-64 gebruiker nauwelijks te verschillen van die van de Nederlandse:
1.Flight Simulator; 2.Eureka(duits); 3.Ghostbusters; 4.Super Huey; 5. Pitstop 2; 6.Summer Games; 7.Give my regards...; 8.Indiana Jones; 9.Baseball; 10.Beach Head.
U ziet het, de computer kent geen grenzen...

Jonge Astronauten

In de Verenigde Staten is Commodore één van de veertien sponsors van het nationale Young Astronauts Program. Dit project is opgezet om bij de jeugd de belangstelling voor de ruimtevaart en de daaraan verwante wetenschappen te stimuleren. Iedere groep van 5 tot 30 scholieren van 7 tot 18 jaar kan zich opgeven en krijgt dan jaarlijks ten minste vier opdrachten voor onderzoek of studie. Daarnaast ontvangen de deelnemers een tijdschrift over het project, er zijn wedstrijden, bezoeken aan ruimtevaartcentra en zij kunnen deelnemen aan Young Astronauts summer camps.
De bijdrage van Commodore bestaat uit het ter beschikking stellen van alle voor het project benodigde hardware met een aantal toepasselijke software pakketten, waaronder SKY TRAVEL en JUST IMAGINE. Het project staat onder leiding van de in Washington gevestigde Young Astronaut Council. Voorlopig kunnen er alleen Amerikaanse scholen aan meedoen maar de Council wil het project naar andere landen, waaronder zelfs de Sovjet Unie, uitbreiden.
Voor nadere informatie: The Young Astronaut Council, P.O.Box 65432, Washington, D.C. 20036, U.S.A.

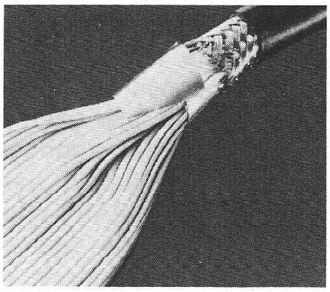
APPLE op 64

Het is een dure oplossing, want een werkende emulatie voor Apple 2 software op de C-64 kost meer dan 2000 gulden, maar voor de liefhebbers is het er nu. Een systeem om Apple software te gebruiken op de 64, dat **SPARTAN** heet en door MIMIC geleverd wordt (1112 Fort str. Fl.6N, Victoria BC Canada V8V4V2).

Video digitizer

In Keulen zagen we ook nog de VD 64 digitizer, waarmee men beelden van videorecorder of Tv kan opnemen en digitaliseren, ofwel omzetten in een computerbeeld. Het werkt via de user-poort en past op de 64, de 128 en de SX64. Het resultaat (opgebouwd uit 160 x 200 punten) wordt in 4 grijstinten vastgelegd, maar kan achteraf met kleuren bijgewerkt worden, met Koalapainter zelfs 16 kleuren. De maximale snelheid is 3 digitale beelden per sekonde en er kan met de 801,802, Epson en HP Inktjet printers gewerkt worden voor afdrukken. Compatibel met het Koalapainter formaat. Leverancier: MERKENS Fuchstanzstrasse 6a, 6231 Schwalbach BRD (0)6196-83030.

Bandkabel



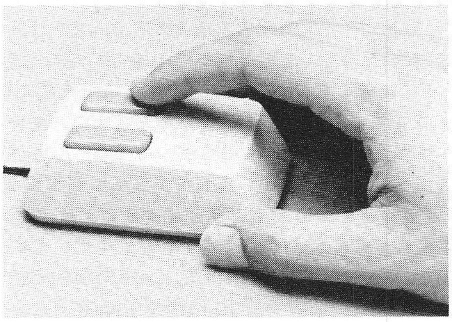
Bij het maken van kabels, bv. voor een RS-232 of Centronics verbinding, moet men kiezen tussen het gemakkelijk monteren van de connectors bij bandkabel en het gemakkelijk wegwerken en leiden van de kabel zelf bij ronde kabel. Ronde bandkabel, zoals deze van 3M Schotchflex, lost dat probleem op. De voordelen van beide kabelsoorten gecombineerd in één type.

Databank training

Van Grolier uit de VS (212-696975) zagen we een trainigsprogramma voor het aanroepen en gebruiken van databanken. Hoewel diensten als **the Source, Compuserve en Dow-Jones** voor computeraars hier wat duur zijn gezien de telefoonverbindingen, is het idee hierachter zeker de moeite waard. Via gesimuleerde verbindingen met een databank leert de potentieële gebruiker daarvan omgaan met aanlog procedures, zoekstructuren, downloading informatie en software, kortom de ins en outs van de (tele-)communicatie met zo'n externe databank. We zouden voòr bv. het onderwijs ook hier wel een dergelijk programma kunnen gebruiken, om het gebruik van Viditel en de diverse bulletin-boards te trainen.

Muis voor C-64

U kent die tafeldiertjes wel, zo'n besturingsgeval met staart en wat knopjes, die als alternatief voor toetsenbord het summum aan gebruiksvriendelijkheid belooft. Gebaseerd op het PARC interface, dat in de Star en de Lisa zijn eerste concrete toepassingen vond. Voor de 128 komt er vast oook een, op de PC-10 zijn er al heel veel soorten te krijgen.

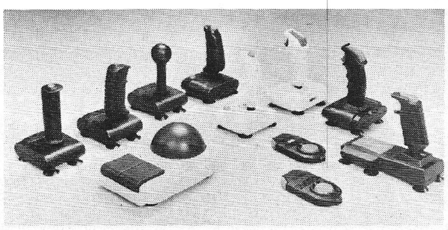


Eindelijk is er een muis voor de Commodore-64 en het ziet ernaar uit dat het een muis wordt die gaat "brullen": het Britse bedrijf SMC heeft nu in haar Connexions lijn een "Magic Mouse" voor de C-64 die primair bedoeld is voor grafische toepassingen. De muis werkt echter ook met vrijwel alle andere programma's en kan dan in plaats van een joy-stick worden gebruikt. Voor doorgewinterde knuppelartiesten zal dat in het begin wel even wennen zijn, maar ja, we moeten natuurlijk allemaal met de tijd meegaan. Met behulp van de muis kunnen 16-kleuren plaatjes naar een disk weggeschreven worden en verdere functies zijn o.a. het creëren van line rays, frame boxes, horizontale en verticale lijnen, cirkels en de mogelijkheid om zeer gedetaileerde inkleuringen te maken. Er zijn er twee soorten en formaten tekst en is er een extra dunne "pen" voor fijn tekenwerk. De sprite designer kan 200 sprites ontwerpen en U kunt ikonen maken om in uw eigen programma's te gebruiken.
Het produkt is ontwikkeld in de U.K., maar door produktie in het Verre Oosten kan SMC de prijs relatief laag houden (59.95 pond voor het hele pakket). Maar denk er aan, zo'n muis werkt alleen maar met daarop afgestemde pakketten. De eerste stap is nu gezet, maar wacht eerst eens af of de standaardpakketten als Superbase en Easyscript dit muis gebruik gaan ondersteunen.

Nuttige stofkap



Bijgaande stofkap is qua vormgeving erg fraai en biedt ook nog plaats aan wat nuttige bureau-artikelen. Pas overigens bij stofhoezen altijd op, wanneer de machine aan blijft staan, kan de warmte soms niet weg en brandt de zaak van binnen door.



Van Spectravideo komen nogal wat soorten joysticks en ander besturingstuig, zoals trackballs.

Epson GX-80

Van de bekende leverancier Epson is er nu ook een budget-printer. Voor minder dan 1000 gulden ex BTW (maar een C-64 interface met alle mogelijkheden kost ƒ 169 extra) is er nu de GX-80 met een snelheid van 100 tekens/sec. Voor briefkwaliteit is de snelheid 20 t/s. Inl. 04139-8883

Karateka

Van Broderbund is er het sport/spelprogramma Karateka. Niet goedkoop met ƒ 99,-, de liefhebber van de oosterse vechtsporten kan zijn hart ophalen. Er zit een heel scenario in, de speler moet de Warlord AKUMA bevechten over goed scrollende tonelen, in een fraaie animatie. Kwaliteit!
inl. 023-319290

Samson Jeugdcursus

Een computercursus op basis van LOGO wordt door Samson gegeven aan kinderen van 8 tot 16 jaar in de vakantie. Voorlopig voor eigen personeel, maar men wil dit ook uitbreiden.
Inl. P van Roon 01720-62158

NABESTELLEN

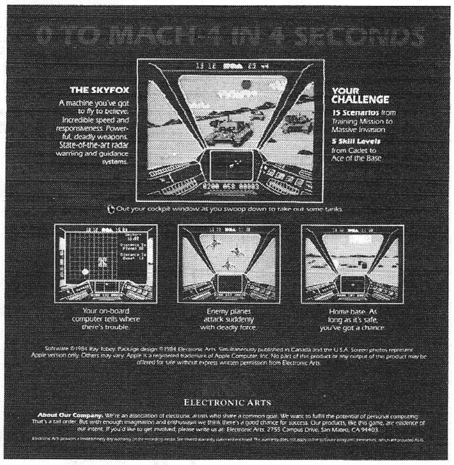
Wie oude nummers van Commodore-Info wil hebben, krijgt door overmaking van ƒ 6,75 per nummer op giro 1585491 tnv. SAC dat opgestuurd. Op andere wijze bestellen is helaas niet mogelijk, maar wel kunt u op de Commodore-Info beurs op 14 september ook oude nummers ter plaatse kopen.

ADRESWIJZIGING
Commodore Info verhuist naar:
Den Texstraat 5a,
1017 XW te
Amsterdam
Tel. 020-273198

SOFTWARE ★ SOFTWARE ★ SOFTWARE







Het aanbod van nieuwe software van vaderlandse bodem is in deze zomerperiode niet erg groot, daarom deze keer voornamelijk programma's, die we in juni van de CES in Chicago in de VS meenamen. Wel hebben we nu Aackotext en Aackobase van Aackosoft in nieuwe verpakking en met een keurig verzorgde handleiding in huis, maar daar komen we een volgende keer op terug. Ook de educatieve Fisher Price software houdt u nog tegoed.

RACING DESTRUCTION SET
Van Electronic Arts
voor C-64 / disk.

Anders dan het vorige spel is dit er weer een uit de goede traditie van Electronic Arts, de mannen van de legosoft-voorloper Pinball Construction Set en basketball-topper One-on-One. De graphics van Racing DS zijn nog niet helemaal het einde, maar toch is dit een zeer opmerkelijk stuk software voor de Commodore 64. Zoals de naam zegt, een race-spel. Maar wel een leuk en leerzaam spel, zelfs als racen niet direkt je voorkeur heeft.
In het spel zitten de mogelijkheden om je eigen racebaan te kiezen uit een keur van meegeleverde parcoursen, al zijn die wel Amerikaans natuurlijk. Daarnaast is het mogelijk, hoe kan het anders bij EA, om je eigen circuit vanaf de basis op te bouwen. En daarbij is de opbouw niet beperkt tot het aangeven van bochten etc., maar ook kun je naar wens een keuze uit maken uit het inbouwen van hellingen, heuveltjes en kuilen, terwijl als laatste ook het wegdek -per segment- kan bestaan uit asfalt, gravel of zelfs sneeuw.
De constructie van de baan kan (na geringe instruktie) zelfs door een 8-jarige worden uitgevoerd, zoals we hebben geconstateerd.
Voordat de race begint , want je moet natuurlijk ook wel de zelfgemaakte weg op om je kunsten te vertonen, kun je een keuze maken uit diverse modellen race/crosswagen en motoren, uiteraard al naar gelang het parcours. Een Formule I is niet erg geschikt voor een cross-circuit. Bij deze optie heb je nog vele submogelijkheden om de wagens naar eigen inzicht op te tuigen met het nodige motorvermogen, bandensoort en max.snelheid etc. zodat het mogelijk wordt om de tegenpartij te voorzien van een ander model/ uitvoering waardoor een handicap kan worden ingebouwd.
Ook de zwaartekracht kan worden ingesteld op 10 a 12 niveau's.
Mijn kinderen en ikzelf hebben met plezier dit spel ondergaan en er meer dan normale tijd aan besteed. De oorzaak hiervan ligt wellicht in het feit, dat er niet alleen door 1 speler tegen de computer kan worden geraced, maar ook (met twee joysticks) 2 spelers het tegen elkaar kunnen opnemen, waarbij je het elkaar goed lastig kunt maken, door botsingen, (GO-CARTs), aanwezige landmijnen en olieplassen en wat al niet meer. Goede stuurmanskunst strekt bij het spelen tot aanbeveling. Als je maar enigszins van dit soort spellen houdt, mag dit 'Racing Destruction Set' zeker niet aan je bibliotheek ontbreken. Vooral de split-screen uitvoering bij racen met 2 spelers, waarbij ieder zijn eigen stukje parcours kan overzien, maar de zaak weer samenkomt wanneer de wagens dicht bij elkaar komen, is heel amusaant. Prima en net even anders spel!

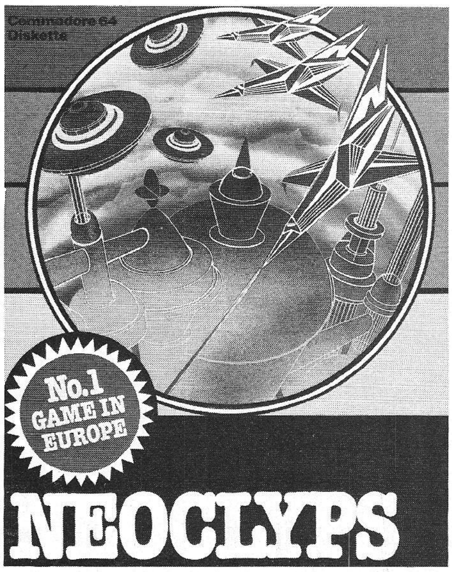
SKYFOX
Van Electronic Arts
voor C-64 / disk.

EA brengt het door Ray Tobey geschreven spel 'Skyfox' voor de C-64 op de markt. Dit stuk software simuleert de cockpit van een supersnelle

gevechtsjet, die (volgens de cover) in staat is om binnen 4 seconden een snelheid van Mach-4 te bereiken.
De bedoeling van dit spel is om met gebruikmaking van de HQ-computer en de aan boord van de straaljager aanwezige hulpmiddelen, zo snel mogelijk alle vijandige legervoer- en vliegtuigen uit te schakelen. Dit spel valt in de categorie van behendige schietspellen, en wijkt in z'n uitvoering alleen maar af, doordat er een ongewoon groot aantal mogelijkheden in de software zitten, die het oefenen op een lager niveau vergemakkelijken. De Hires-schermen zijn aardig, wel vrij langzaam en weinig gedetailleerd.
Als je al beschikt over de nodige aktie/schiet-software, zul je 'Skyfox' zeker niet missen, we waren er niet stuk van.

NEOCLIPS
Van BCI-Software
voor C-64 / disk.

Van geheel ander kaliber is 'Neoclips', een disk (met 8-regelig stencil voor het gebruik) in een grote doos, waarvan het enige bijzondere de tekst : 'Number 1 game in Europe' te melden is.
Ik heb in Nederland nog weinig van dit spel vernomen, en als het aan mij ligt, hoeft dat eigenlijk ook niet.
Het spel is een Arcade-game, met de onvermijdelijke Bad Guys, die van alles willen vernielen. Om dat tegen te



houden kun je met je 'Good guy-vliegmachien alles in de puin schieten, tot jezelf wordt geraakt.
Mooie namen, grote kreten, maar weinig meer dan al die andere arcade-spelletjes, waar we al zoveel van bezitten. Er zijn veel betere op de markt. Hoogstens mag de prijs, dit zijn 5 dollar spelletjes in de VS, terwijl voor de EA spellen 30, 40 of meer dollars moeten worden neergeteld, een excuus vormen.

SFX MUSIC MAKER
Van Commodore
Voor C-64 / cassette (of disk) / mechanische uitbreiding.

Een stukje software en een klein plastic toetsenbordje, dat over de standaard C-64 moet worden geschoven, kun je voor rond 100 gulden het jouwe noemen, samen met een zing-boek, waarin enkele notenbalken met tophits uit vervlogen dagen zijn gedrukt. Het is de bedoeling om op deze manier de computer als synthesizer te kunnen gebruiken, en naar de letter, is Commodore daarin geslaagd. Nadat de software (via Novo-load) is ingeladen, krijg je een door F-toetsen bestuurd menu, waardoor je van de verschillende mogelijkheden gebruik kunt maken.
De schermopmaak is verder van weinig belang, want het gaat hier om geluiden en muziek. Door op de plastic toetsjes te drukken, komt er geluid uit de TV (of, beter nog de HIFI) dat doet denken aan een simpel electronisch orgeltje. De toetsomvang van het klavier is 2 oktaaf. Dit ding is te klein om er goed op te kunnen spelen, maar daardoor geschikt voor kinderhandjes.
Er zitten in totaal 8 voor te programmeren stemmen in gebouwd, die stuk voor stuk ten gehore kunnen worden gebracht, al of niet met een meespelend ritme en zelfs nog een baspartij. Naast de sequencer (een moduul, waarbij je eerst een deun kunt intoetsen, dan korrigeren en daarna laten afdraaien) kun je de liedjes die je speelt uiteraard voor later gebruik wegschrijven naar tape of disk.
Kennis van het notenschrift is niet direkt vereist, omdat het instruktieboek, via 'cdefgabc' en stickertjes, duidelijk maakt waar de toetsen op het klavier zitten.
De 3 ritme-opties klinken leuk, maar zijn wel steeds hetzelfde, dus blijven de klanken beperkt door de onophoudelijke dreun van de ritmesectie, temeer daar de verhouding tussen beide niet valt te regelen. Je hebt een grote kans snel dit spel snel te hebben uitgehoord, meegeholpen door het feit, dat na korte of lange tijd het spel uit zichzelf al stopt met de schermboodschap 'out of memory in 24563'. Niet zo'n beste beurt voor een exempel van Commmodore's eigen merk.

STRIPPOKER
Van Artxwork

Voor iedereen die er altijd al van heeft gedroomd, nu eens echt met een pokerface z'n 'full house' te kunnen openleggen, heeft Artworx kennelijk gemeend een extra dimensie aan het pokerspel te moeten toevoegen, door er een soort Hires striptease van te maken.
Nu weet ik wel, dat er genoeg aardigs van strippoker valt te zeggen, en ook dat er niet altijd mogelijkheden zijn om dit spel te kunnen spelen, maar m.i. ligt het accent daarbij niet op het pokeren.
In de electronische versie van strippoker maken we kennis met twee gastvrouwen, die de welluidende namen van Suzy en Melinda hebben meegekregen. De speler krijgt $100 in z'n hand en kan daarna een keuze uit de beide aanlokkelijke dames maken, om het spel mee te spelen.
Daarna gebeurt er een poos niets, en vervolgens verschijnt Suzy (of Melinda) ten voeten uit op het scherm. Wat er verder gebeurt, is niet moeilijk te raden, want het doel wordt nu natuurlijk, zo snel mogelijk de dame uit de bloes en iedere $100 is weer een kledingstuk. De hires voorstelling van de dames is allesbehalve optimaal, dus daarvoor hoef je dit spel niet aan te schaffen. Wel kun je nog het een en

Vervolg Software

ander leren over de enorme hoeveelheid dames-lingerie die er blijkt te bestaan. Geen aanrader, maar de dames kunnen aardig pokeren.

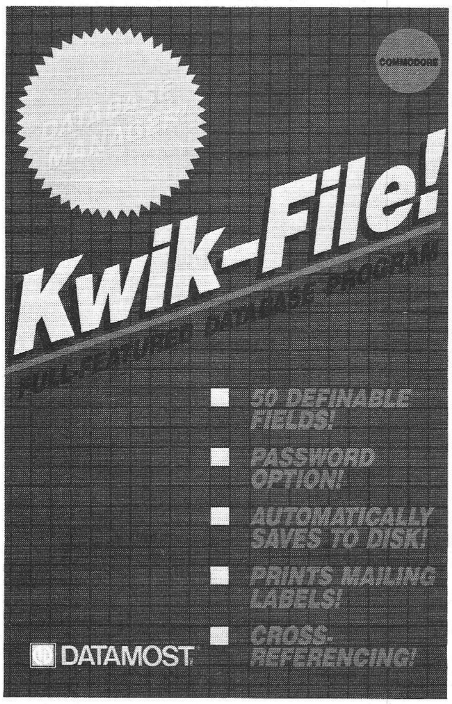
KWIKFILE
Van Datamost
voor C-64 / disk.

Kwik-file moet worden gezien als een serieuze poging om een (betaalbare) imitatie te zijn van b.v. Superbase en wordt samen met wat andere "snelle" programma's in de VS heel goedkoop op de markt gebracht, de zgn. "Budget" software.

Om de prijs hoef je dit database programma dųs niet te laten liggen, maar ondanks de grootse beloften op de doos en in het instruktieboek, komt er weinig meer uit dit bestand dan iets, waarvan wij in dit blad al listings hebben gepubliceerd.

Uiteraard is het niet fair het hiermee af te doen, want 'Kwik-file' heeft ook z'n goede kanten. Zo kun je maximaal 50 veldèn per record definiëren en dat laat veel ruimte voor ingewikkelder bestanden, je kunt -per bestand- een wachtwoord inbouwen, labels van de data-gegevens kunnen worden geprint (op een vreemd sticker formaat!?) en ook een cross-referentie hoort bij de opties.

Het Engelstalige handboek geeft voldoende informatie over het gebruik van deze data-base, maar je moet het hele boekje wel doornemen, voordat je begint met het serieus aanleggen van bestanden. Het gebruiken van 'Kwik-file' is vrij gemakkelijk, omdat er voor iedere handeling een keuze-menu op je scherm komt.



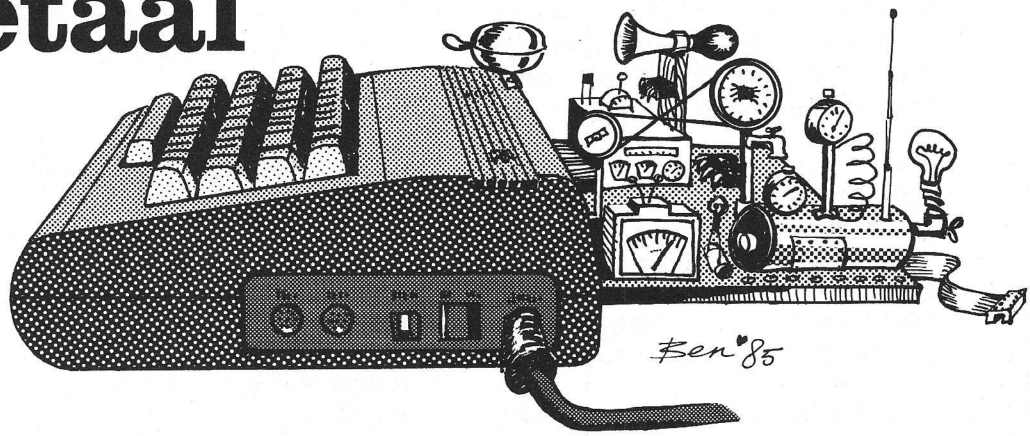
Kortom, voor de beginnende database-bouwer een leuk programma, maar verwacht geen wonderen van 'Kwikfile'. Voor commerciele toepassingen heeft dit programma weinig waarde.

J.B. ●

Deze aflevering gaat ook weer over de 65xx-processor, het hart van de Vic en de 64. We zullen ons hierin bezig gaan houden met rekenen en interrupts. Het programma van deze keer laat zien hoe je de "1/60 seconde interrupt" heel makkelijk voor eigen doeleinden kunt gebruiken.

Programmeren in machinetaal

door Sjoerd Bakker



Een computer is maar een beperkt wezen, dat zal u wel eens zijn opgevallen. Maar wist u, dat zelfs zijn "zogenaamde" rekenkracht maar gebaseerd is op de allereenvoudigste principes en dat bv. vermenigvuldigen en delen onbekende begrippen zijn in het hart van de micro. Dergelijke functies moeten worden opgebouwd uit nog eenvoudiger handelingen.

Aflevering 4:

De instrukties *nader bekeken*

Rekenen

De instruktieset, dus de reeks commando's waarop de processor reageert, kent maar twee rekenkundige bewerkingen die we ook gebruiken in het decimale stelsel, namelijk optellen en aftrekken. Alle andere funkties, zoals vermenigvuldigen en delen, moeten in principe met behulp van deze twee door de programmeur worden opgebouwd. Een vermenigvuldiging als 5 x 2 wordt bijvoorbeeld opgelost door hem eerst te vertalen in een andere som: 2+2+2+2+2. In principe, omdat de processor een aantal andere tweetallige bewerkingen kent, waarvan je meestal beter gebruik kunt maken als je gaat rekenen in machinetaal.

De instrukties voor optellen en aftrekken zijn: ADC en SBC. Dit zijn afkortingen voor AdD with Carry en SuBtract with Carry; vrij vertaald met respectievelijk:

ADC : Tel bij de inhoud van de Accumulator de inhoud van de geadresseerde geheugenplaats op en vermeerder het resultaat met de waarde van het Carry-bit.

SBC : Trek van de inhoud van de Accumulator de inhoud van de geadresseerde geheugenplaats af en verminder het resultaat met 1 als het Carry-bit 0 is.

En daarmee zijn we weer volop in de geheimtaal, even een korte herhaling voor wie de vorige artikelen miste. De Accumulator is een register in het geheugen, een soort hulpgeheugentje, dat bij allerlei bewerkingen gebruikt wordt.

Zoals we in de vorige aflevering hebben gezien wordt het C-bit of Carry-bit 1 gemaakt als het resultaat van een optelling groter is dan $FF, oftewel: als het resultaat niet meer op twee hexadecimale cijferplaatsen kan worden ondergebracht wordt de overloop, zeg maar het "één onthouden", op 1 gezet. Is dit wel het geval, dan wordt het C-bit 0 gemaakt. Bij een aftrekking wordt het C-bit 0 gemaakt als het resultaat kleiner is dan $00. Is het resultaat groter dan of gelijk aan $00, dan wordt het C-bit 1 gemaakt. Het nut van deze eigenschappen wordt duidelijk met de volgende voorbeelden.

We willen de getallen $FEFF en $0001 bij elkaar optellen en doen dat met de beperkte commando's die we kennen, terwijl we ook de plaatsjes, waar het resultaat komt te staan, natuurlijk moeten aangeven.

Het resultaat zetten we in de bytes (op de adressen) $033A, $033B en $033C.

Voor je begint aan een optelling moet je het C-bit 0 maken omdat je bij de eerste bytes (de Lo-bytes) natuurlijk nooit een carry hebt.

```
CLC
LDA #$FF             Lo-byte getal 1
ADC #$01             Lo-byte getal 2
STA $033A          Lo-byte resultaat
LDA #$FE             Hi-byte getal 1
ADC #$00             Hi-byte getal 2
STA $033B          2e byte resultaat
LDA #$00
BCC #$02                 Geen carry?
               Dan springen. A=$00
LDA #$01       Wel een carry. A=$01
STA $033C          Hi-byte resultaat
```

Een analyse:

$FF+$01 geeft als resultaat $00 en een "1" C-bit.

$FE+$00 geeft als resultaat $FF (carry !) en een "0" C-bit.

Het resultaat van $FEFF+$0001 is dus $(00)FF00.

Voor aftrekken nemen we als voorbeeld $FF00-$0001.

Deze keer past het resultaat altijd in twee bytes. We nemen hiervoor $033A en $033B. Aan het begin van een aftrekking moet het C-bit altijd 1 worden gemaakt, want voor de eerste bytes (de Lo-bytes) kun je nooit al een "lening" of overloop hebben.

```
SEC
LDA #$00          Lo-byte getal 1
SBC #$01          Lo-byte getal 2
STA $033A         Lo-byte resultaat
LDA #$FF          Hi-byte getal 1
SBC #$00          Hi-byte getal 2
STA $033B         Hi-byte resultaat
```

Een analyse:

$00-$01 geeft als resultaat $FF en een "0" C-bit.

$FF-$00 geeft als resultaat $FE (carry 0 !) en een "1" C-bit.

Het resultaat van $FF00-$0001 is dus $FEFF.

Op deze manier kun je onbeperkt grote getallen bij elkaar optellen of van elkaar aftrekken. Je moet er dus alleen voor zorgen dat je, net als in het decimale stelsel, begint met de twee Lo-bytes en eindigt met de twee Hi-bytes van een getal.

Tweetallige bewerkingen

De 65xx-processor kent twee soorten tweetallige bewerkingen en hoewel dit allemaal in wezen Boleaanse wiskunde is, proberen we het hier eenvoudig te houden.:

❶ AND, OR en EOR
❷ Rotaties en shifts.

1.a AND

De bewerking AND (en denk maar aan optellen als geheugensteuntje) van twee getallen houdt in, dat een bit van het resultaat 1 wordt als van beide getallen het overeenkomstige bit 1 is. Anders wordt het resultaatbit 0.

0 AND 0 = 0
0 AND 1 = 1
1 AND 0 = 0
1 AND 1 = 1.

1.b OR

De bewerking OR (en denk maar aan OF als hulpje) van twee getallen houdt in, dat een bit van het resultaat 1 wordt als van minstens een van beide getallen het overeenkomstige bit 1 is. Anders wordt het resultaatbit 0.

0 OR 0 = 0
0 OR 1 = 1
1 OR 0 = 1
1 OR 1 = 1.

1.c EOR

De bewerking EOR van twee getallen houdt in, dat een bit van het resultaat 1 wordt als van precies een van beide getallen het overeenkomstige bit 1 is. Anders wordt het resultaatbit 0.

0 EOR 0 = 0
0 EOR 1 = 1
1 EOR 0 = 1
1 EOR 1 = 0

De bewerkingen OR en EOR lijken dus op elkaar, alleen als er twee énen zijn is het resultaat anders. In wezen geven deze drie commando's dus alledrie een resultaat op basis van vergelijking van 2 andere bits aan.

2.a ROTATIES

Bij een rotatie, weer zo'n woord dat een beetje verminkte betekenis heeft, verschuiven alle bitwaardes van een getal een cijferplaats naar links of rechts. Vervolgens wordt de cijferplaats waarheen nog geen bitwaarde is verschoven gelijk gemaakt aan de waarde van het C-bit en wordt het C-bit gelijk gemaakt aan de "laatste" bitwaarde die moest worden verschoven.

2.b SHIFTS

Ook bij shifts verschuiven alle bitwaardes van een getal een cijferplaats naar links of rechts. De cijferplaats waarheen nog geen bitwaarde is verschoven wordt nu echter altijd gelijk gemaakt aan 0. Het C-bit wordt weer gelijk gemaakt aan de laatste bitwaarde die moest worden verschoven.

Omdat de bits van een tweetallig getal van rechts naar links verdubbelen in waarde (128,64,32,16,8,4,2,1), betekent een linkse shift een vermenigvuldiging met 2 en een rechtse shift een deling door twee. En dat is natuurlijk al een aardig snelle manier van rekenen.

Interrupts

Het beinvloeden van de computer van buitenaf gaat via interrupts (onderbrekingscommando's), dat zijn ahw. de signalen waar de computer steeds naar kijkt om te weten of hij iets anders moet gaan doen.

De meeste computerschaakprogramma's produceren een pieptoon als ze een zet hebben gedaan. Dat is maar goed ook, want anders zou de menselijke tegenstander om de 10 minuten moeten controleren of de computer al klaar was. Het handelen van een mens na zo'n pieptoon vertoont veel overeenkomsten met het handelen van de microprocessor na een interrupt. Je stopt even waar je mee bezig was, bijvoorbeeld het lezen van Commodore-Info, je typt een nieuwe zet in de computer, en gaat weer verder met lezen. Een interrupt kunnen we dus omschrijven als een forcering van een tijdelijke onderbreking (het intypen van een nieuwe zet) van het hoofdprogramma (het lezen van Commodore-Info). Vaak is er sprake van een vast tijdsverloop tussen twee interrupts.

De 65xx-processor is voorzien van 3 verschillende interruptingangen. Ze heten **Reset, NMI, en IRQ.**

Als er op een van deze ingangen een stroomstoot wordt gegeven, maakt de processor de opdracht waarmee hij bezig was af, en springt daarna onmid-

(illustratie 1)

($ØF) ØØØØ1111	($ØF) ØØØØ1111	($ØF) ØØØØ1111
($55) Ø1Ø1Ø1Ø1 AND	($55) Ø1Ø1Ø1Ø1 OR	($55) Ø1Ø1Ø1Ø1 EOR
($Ø5) ØØØØØ1Ø1	($5F) Ø1Ø11111	($5A) Ø1Ø11Ø1Ø

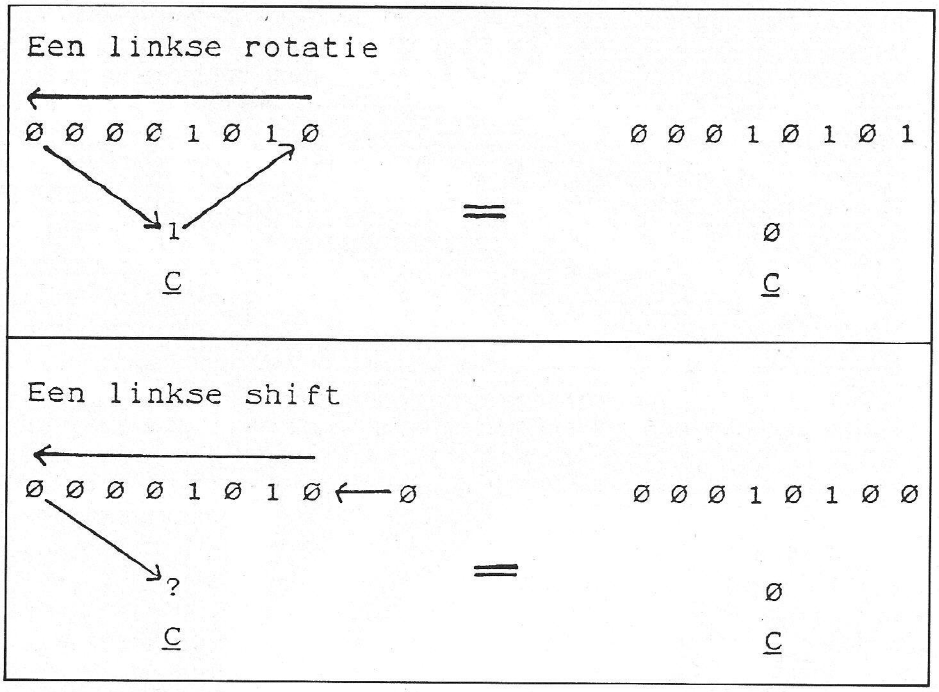
dellijk naar de betreffende interrupt-routine.

De Stack

Voor we gaan bekijken hoe een interrupt nu precies in zijn werk gaat, moeten we eerst iets meer weten over de Stack. De Stack zou je kunnen vergelijken met een wat primitieve garderobe. De jassen worden er niet opgehangen, maar op een grote stapel gegooid. De bovenste jas moet er dus altijd worden afgehaald door degene die hem het laatst heeft afgegeven. In computertaal noemen we een zodanig gestruktuureerde stapel een LIFO (Last In First Out).

De stack is een stapeltje gegevens in het processorgeheugen, waar de processor allerlei nuttige gegevens even wegzet om ze later weer op te pakken. Zo gebruikt de processor de Stack om er bij een interrupt het eerstvolgend adres van het hoofdprogramma (Hi-byte eerst) en de inhoud van de registers op te zetten. (Bij JSR, Jump to Subroutine saving Returnadres, zet hij alleen het eerstvolgend adres van het hoofdprogramma op de Stack.) Vervolgens wordt het I-bit 1 gemaakt: interrupts over de IRQ-lijn worden verboden. Dan springt hij naar de interrupt- of subroutine. Als hij dan uiteindelijk een RTI of een RTS retourcommando tegenkomt haalt hij het bewaarde adres (en bij een RTI ook de inhoud van de registers) weer van de Stack, en vervolgt het hoofdprogramma vanaf het punt waar hij gebleven was. De programmeur hoeft voor dit alles trouwens niet te zorgen: het gebeurt helemaal hardwarematig.

Voor de Stack zijn de geheugenplaatsen $0100 t/m $01FF gereserveerd. Omdat de Hi-byte van elke Stackplaats dus gelijk is aan $01, worden ze geadresseerd met alleen de Lo-byte. Dit wordt gedaan met behulp van de Stackpointer, die altijd wijst naar het eerstvolgende vrije Stackadres. Dat is niet, zoals je zou verwachten het adres+1 van de laatste waarde die we op de Stack hebben gezet, maar het adres-1 van die laatste waarde. Als gevolg hiervan begint de Stack op $01FF en eindigt hij op $0100. Ook het verlagen en verhogen van de Stackpointer gaat geheel automatisch. Met de instruktie TXS is hij ook softwarematig te veranderen, maar dat zou ik de lezers van Commodore-Info niet willen aanraden. Voor je het weet loopt je hele programma in de soep.



(illustratie 2)

Zoals gezegd heeft de 65xx-processor 3 verschillende interruptingangen. We zullen ze alle drie even apart doornemen.

1. Reset

Na een signaal op de Reset-lijn springt de processor naar het adres dat gevormd wordt door de inhoud van $FFFC (Lo-byte) en $FFFD (Hi-byte). In de Kernal-ROM is op deze geheugenplaatsen de waarde $FCE2 ingebakken. Op dit adres begint een routine die o.a. de lokaties $0000 - $0400 vult met beginwaardes voor het operating-system en de mededeling :
****COMMODORE 64 BASIC V2 ... etc. op het scherm zet.

Als de geheugenplaatsen $8004 t/m $8008 echter achtereenvolgens $C3, $C2, $CD, $38 en $30 (ASCII-codes voor CBM80) als inhoud hebben, wordt deze routine na enkele opdrachten afgebroken. Er wordt dan gesprongen naar het adres dat gevormd wordt door de inhoud van lokaties $8000 en $8001. Dit is het geval als er een cartridge in de uitbreidingspoort zit. Cartridges leggen namelijk beslag op de geheugenplaatsen $8000 - $A000. Je kunt natuurlijk ook zelf zo'n cartridge simuleren door bovenstaande waardes in de betreffende RAM-lokaties te poken.

De Commodore 64 wordt ongetwijfeld het meest gereset door hem aan te zetten. Een interne reset vormt de laatste stap van de inschakelprocedure. Deze manier van resetten is echter in vele gevallen ongewenst omdat je door zelfs de kortste stroomuitval al je RAM-geheugen kwijt raakt.

Meestal kun je beter gebruik maken van de mogelijkheid om een resetknop te plaatsen op de uitbreidingspoort of userpoort. Deze hebben beide een reset-pin die via een lijn is verbonden met de resetingang van de processor. Overigens worden deze pinnen in de praktijk verder vrijwel nooit gebruikt als ingang, maar juist als uitgang om aangesloten apparaten te resetten.

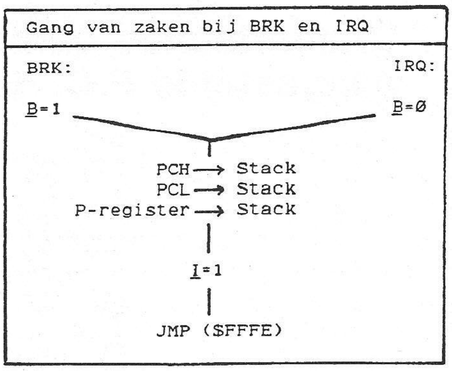
2. NMI

Na een signaal op de NMI-ingang springt de processor naar het adres dat de inhoud vormt van lokaties $FFFA en $FFFB. In de Kernal is dat adres $FE43. Deze routine heeft als tweede opdracht weer een indirekte sprong. Deze keer naar het adres dat de inhoud vormt van lokaties $0318 en $0319. Normaal gesproken zal deze vektor (=pointer die door het operating system zelden veranderd zal wor-

den) wijzen naar de 3e opdracht van de routine. Maar omdat de vektor in RAM ligt kan deze door de gebruiker worden veranderd.
Vervolgens wordt er gekeken of de NMI intern veroorzaakt werd (door CIA 2). Was dit niet het geval, dan wordt er getest op "CBM80" in $8004 t/m $8008. Zo ja, dan wordt er gesprongen naar het adres dat de inhoud vormt van $8002 en $8003. Zo nee, dan werd de NMI veroorzaakt door het gelijktijdig indrukken van run/stop en restore op het toetsenbord en wordt er een BASIC Warmstart uitgevoerd. Aan het eind hiervan keert de computer terug in "direct BASIC-mode".
De meest voorkomende NMI is het indrukken van run/stop in kombinatie met restore. Negen van de tien commerciele programma's op cassette of disk maakt namelijk gebruik van de mogelijkheid om een cartridge te simuleren. Daartoe schrijft het programma zichzelf (e.v.t. na een run) weg naar de geheugenplaatsen rond $8000. In kombinatie met een automatische start vormt zo'n nep- cartridge een bekende methode om het krakers wat moeilijker te maken. Als je nu run/stop en restore indrukt kom je meestal terug aan het begin van het programma. Resetten geeft een soortgelijk effekt.

3. IRQ

Na een signaal op de IRQ-pin springt de processor naar het adres dat de inhoud vormt van $FFFE en $FFFF. In de Kernal is dat $FF48. Hier wordt gekeken of de IRQ door een machinetaalprogramma (instruktie BRK) werd veroorzaakt of niet. In het eerste geval wordt er gesprongen naar het door vektor $0316/17 aangegeven adres. Bij gebruik van een machinetaalmonitor of assembler wijst deze vektor meestal naar een subroutine die eindigt met de .R- instruktie (inhoud registers laten zien). Hierdoor kun je onderdelen van machinetaalprogramma's testen door ze telkens te laten eindigen met een BRK.
Als de IRQ niet werd veroorzaakt door een BRK wordt er gesprongen naar vektor $0314/15. Deze is normaal gesproken gelijk aan $EA31. De hier be-



illustratie 3)

ginnende routine wordt bij werken in BASIC 60 keer per seconde aangeroepen om de tijd te verhogen (TI$) en om te controleren of er ook toetsen zijn ingedrukt.
Het zou te ver gaan om hier alle bronnen te gaan behandelen die een IRQ kunnen veroorzaken.
Een belangrijk punt nog: het verbieden van interrupts. Resets kunnen in geen geval worden verboden. NMI's kunnen alleen per bron in de CIA 2 worden verboden. IRQ's kunnen zowel per bron in de CIA 1 en de VIC 2, als in de processor worden verboden. Dat laat-

ste gebeurt met de instruktie SEI (SEt Interrupt disable status). Interrupts worden weer toegestaan met de CLI (CLear Interrupt disable status).

Met SEI worden dus alle IRQ's en BRK in een keer verboden. Dat wil niet zeggen dat er geen IRQ of BRK gegeven kan worden, maar alleen dat de processor deze negeert. Mocht er tijdens een SEI- conditie een IRQ zijn opgetreden, dan wordt deze direkt na een CLI of RTI alsnog uitgevoerd, tenzij de IRQ-flag van de betreffende chip eerst werd teruggeset! Hou hier terdege rekening mee.

P-register

We komen nu terug op de nog niet behandelde bits van het P- register: de bits B, D, I, en V.
Het B-bit (Break) wordt direkt na een IRQ 0 en na een BRK 1 gemaakt. Door dit bit te testen weet de processor of er naar de IRQ-, dan wel naar de BRK-routine moet worden gesprongen.
Het D-bit (Decimal) dient om optellen en aftrekken in BCD (Binary Coded Decimal) -formaat mogelijk te maken. Het D-bit moet dan 1 worden gemaakt. In de praktijk wordt dit vrij weinig gedaan, en ik zal dan ook niet op het hoe en waarom van deze methode ingaan. Belangstellende verwijs ik naar "Programming the 6502" (de "Bijbel" van de 65xx-processor) van Rodnay Zaks. Er bestaat ook een Nederlandse vertaling van.
Het I-bit (Interrupt) dient om IRQ's en BRK's toe te staan (CLI) of te verbieden (SEI). Na iedere interrupt wordt het I-bit automatisch 1 gemaakt om tijdens de interrupt-routine niet weer door een interrupt gestoord te worden. Na een RTI wordt het I-bit automatisch 0 gemaakt.
Het V-bit heeft alleen een funktie bij rekenen volgens de "Two's Complement-methode". Ook hier krijgt de hobby-programmeur zelden mee te maken, en verwijs ik dus weer naar het boek van Zaks.

Instrukties

Voor de volledigheid zullen we alle 56 65xx-instrukties nog eens afzonderlijk gaan bekijken. De meeste zijn we in de loop van deze serie al een keer tegengekomen, dus daar zullen we niet al te lang bij stil blijven staan. Zie voor volgorde en groepsindeling het schema uit de vorgie aflevering.

Deze aflevering was de laatste, die specifiek over de 65xx-processor ging. Ik realiseer me, dat de informatie die erin werd gegeven op velen een nogal "droge" indruk zal hebben gemaakt. Machinetaal staat in verhouding tot BASIC als een bit tot een hires-scherm: nuttige saaiheid tegenover onzelfstandige glamour. En de meeste mensen voelen zich nu eenmaal meer aangetrokken tot het laatste dan tot het eerste, wat hun goed recht is.
Aan een taal heb je pas wat als je in het buitenland komt, dat wil zeggen: als je hem in de praktijk kunt toepassen. Daarom zullen we ons in de volgende afleveringen bezig gaan houden met het programmeren van de Commmodore 64-chips voor beeld, geluid en input/output.

S.B

Programma

Het toetsenbord is via 2x8 lijnen verbonden met registers $DC00 en $DC01 van de CIA 1-chip. Om te controleren of er ook toetsen zijn ingedrukt, roept de Commodore 64 in BASIC-mode 60 keer per seconde een routine aan die o.a. de inhoud van deze registers bekijkt. Dit gebeurt met behulp van een timer-interrupt. Als we nu de inhoud van $Ø314 en $Ø315 gelijk maken aan het begin van een andere routine (zie IRQ), kunnen we deze timer-interrupt zonder veel moeite benutten voor eigen doeleinden. Het programma van deze keer doet dit in "regels" $9029 t/m $9030.

Het doel hiervan is om elke 60e second een bit van een hires-scherm "aan te zetten". Niet zo spectaculair, maar wel een eenvoudig en effektief voorbeeld van interruptprogrammering. Let ook op het gebruik van tweetallige bewerkingen EOR, ROL en ROR. Een analyse van dit programma volgt nog.

```
10 a=36864
20 for x=0 to 79: read y
30 poke a+x,y: next: sys a
100 data120, 32, 68,229,160,128,132,254
110 data160,  0,132,251,162, 32,134,252
120 data152,145,251,200,208,251,230,252
130 data165,252, 42, 16,243,134,252,169
140 data 28,141, 24,208,169, 59,141, 17
150 data208,169,144,141, 21,  3,169, 63
160 data141, 20,  3, 88,144,254,102,254
170 data200,208,249,230,252,144,245,173
180 data 13,220,177,251, 69,254,145,251
190 data104,104,104, 40,102,254,  8, 64
```

```
., 902b 8d 15 03 sta $0315
., 902e a9 3f    lda #$3f
., 9030 8d 14 03 sta $0314
., 9033 58       cli
., 9034 90 fe    bcc $9034
., 9036 66 fe    ror $fe
., 9038 c8       iny
., 9039 d0 f9    bne $9034
., 903b e6 fc    inc $fc
., 903d 90 f5    bcc $9034
., 903f ad 0d dc lda $dc0d
., 9042 b1 fb    lda ($fb),y
., 9044 45 fe    eor $fe
., 9046 91 fb    sta ($fb),y
., 9048 68       pla
., 9049 68       pla
., 904a 68       pla
., 904b 28       plp
., 904c 66 fe    ror $fe
., 904e 08       php
., 904f 40       rti
```

```
., 9000 78       sei
., 9001 20 44 e5 jsr $e544
., 9004 a0 80    ldy #$80
., 9006 84 fe    sty $fe
., 9008 a0 00    ldy #$00
., 900a 84 fb    sty $fb
., 900c a2 20    ldx #$20
., 900e 86 fc    stx $fc
., 9010 98       tya
., 9011 91 fb    sta ($fb),y
., 9013 c8       iny
., 9014 d0 fb    bne $9011
., 9016 e6 fc    inc $fc
., 9018 a5 fc    lda $fc
., 901a 2a       rol
., 901b 10 f3    bpl $9010
., 901d 86 fc    stx $fc
., 901f a9 1c    lda #$1c
., 9021 8d 18 d0 sta $d018
., 9024 a9 3b    lda #$3b
., 9026 8d 11 d0 sta $d011
., 9029 a9 90    lda #$90
```

Op hardwaregebied, dus insteken, aansluiten, optuigen of connecten, blijft er van alles in beweging, de uitbreidingen en modules vliegen je nog steeds om de oren.

Eprom-Bank



HARDWARE

Zo kregen we kort geleden nog wat nieuwe hardware uitbreidingen voor de C-64 onder ogen, die zeker het vermelden waard zijn. Een ervan is de 'Auto start Eprom Bank', die geleverd wordt door R.A.D. Computer & Electronic Supplies in Amsterdam.

EPROM BANK

Deze print (aansluiting expansiepoort) kost *f* 125.- en opent voor de serieuze gebruikers een scala van nieuwe mogelijkheden.
Standaard is er op deze print een RESET-toets en 1 PROM geplaatst, waarnaast de mogelijkheid bestaat om maximaal 5 (E)PROMS van 4K of 8K op de C-64 aan te sluiten. Eproms zijn geheugenchips, die je zelf kunt volladen met programma's, die er dan permanent (tot je ze zelf weer wist) in blijven staan. Zelfmaak insteekmodules zou je zo'n Eprombank kunnen noemen.
Deze bank van EPROMS is geschikt om gekochte of zelfgeschreven programma's in zowel Basic als machinetaal te runnen, zonder deze eerst van tape of disk in te laden. Een grote vooruitgang, zeker als je bedenkt dat b.v. Simon's Basic etc. via deze Eprombank automatisch tot de standaard uitrusting van de C-64 kan gaan behoren. De PROM bevat een menu, waardoor bij het opstarten een keuze kan worden gemaakt uit de op EPROM aanwezige software. Basicprogramma's starten direkt. Er is voldoende ruimte aanwezig om ook nog op een normale manier met de computer te kunnen werken, zonder dat de print verwijderd hoeft te worden.
De nederlandse handleiding geeft duidelijke aanwijzingen voor het gebruik. Daarin kun je tevens lezen, hoe je te werk moet gaan om je eigen programmatuur, beveiligd en zelfstartend, op deze bank te krijgen. Daarvoor heb je echter wel een EPROM-programmer nodig en die moet je apart aanschaffen.
Enkele feiten: elk programma (tot max.23K) bezet minimaal 1 EPROM-plaats. ROM-programma's zelfstartend (i.p.v. Menu). De print is uit te breiden tot 56K! Voor Simon's Basic op EPROM is een aparte versie nodig. De print is niet al te zorgvuldig uitgevoerd. Software voor de EPROMS en de EPROM-programmer zijn te koop bij dezelfde leverancier. De print past helaas niet goed op de SX-64.

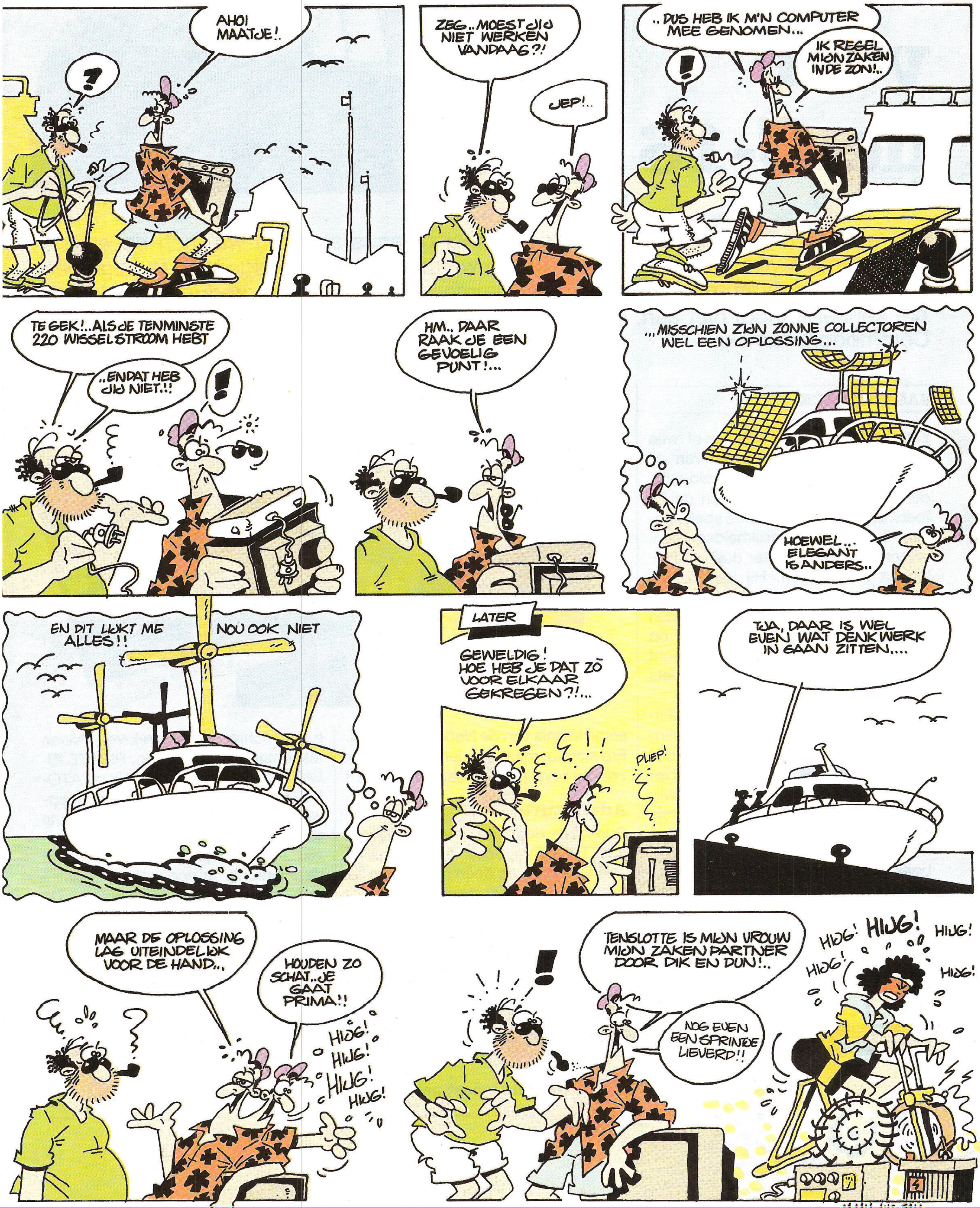
The Final Cartridge.

H.&P. Home en Personal Computers uit Rotterdam brengt een uitbreiding op de markt, waarvan wordt beweerd, dat je verder niets meer nodig hebt, om alles uit de C-64 te halen wat er in zit. Het systeem bestaat uit een unit die op de expansiepoort moet worden aangesloten, en daarnaast een (Centronics) printerkabel, die geschikt is gemaakt om op de userpoort te pluggen. Het voornaamste van deze final cartridge is dan wel het nieuwe operatingsysteem waarop de 64 z'n werk gaat doen.
De cartridge bevat een 'logic board' met 8 IC's en 16K ROM, waarop dit totaal nieuwe operating systeem. De oude Kernal van de 64 wordt niet meer gebruikt, maar volgens H&P runt 99.9% van alle C-64 software op dit O.S. Helaas wordt er niet bij vermeld welke programma's nu juist niet werken. De unit voegt maar liefst 20K RAM aan het Basic gedeelte toe, waardoor je de beschikking krijgt over 63000 Bytes. Verder een goed werkend Centronics printerinterface, waarmee ook alle Commodore tekens op de meeste printers kunnen worden uitgeprint. Dit interface werkt met alle bekende printers, nou ja vrijwel alle. Wel is het bij bepaalde printers helaas nodig vooraf enkele pokes in te typen. Dit betekent, vrees ik, dat bij commerciële software ook deze interface niet altijd zal werken. Printers die geschikt zijn voor Hires-dumps kunnen met hulp van deze uitbreiding daartoe via 1 toets worden geaktiveerd. De meegeleverde (nederlandse) beschrijving geeft voor een paar printertypes voorbeelden hoe je te werk moet gaan.
Naast deze nieuwigheden beschikt deze uitbreiding over de min of meer standaard toolkit-commando's zoals snelladen/saven, DOS 4.0, waarbij nu ook DLOAD en DSAVE, APPEND en CATALOG, de bekende programmeerhulpen en de machinetaal-monitor.
Al met al een aardig stuk werk, waarvoor je *f* 198,– moet betalen. Deze Final Cartridge is voor hen, die moeite hebben met het aansturen van hun alledaagse printers op de niet alledaagse C-64 aansluiting wel interessant. Bij aankoop zou ik willen adviseren om je eigen software en printer eerst met deze unit te proberen, zodat je achteraf niet voor verrassingen komt te staan.

Deze module is aantrekkelijk, zeker omdat de prijs niet ongunstig afsteekt tegenover andere interfaces met dezelfde mogelijkheden. •

SOFT WIR WAR

DOOR BERT TIER.



Software voor de C-16



Het was al haast een illusie geworden, maar onlangs stootten we toch op enkele programma's voor de C-16. Ook bekeken we een Plus/4 cartridge. Om te beginnen waren alle programma-cartridges van Commodore zelf. Andere producenten durven het kennelijk (nog) niet aan, software te ontwikkelen voor deze "zwart" take van de Commodore-familie.



JACK ATTACK

Een spel in arcadestijl voor een of twee personen. Na het inpluggen van de spelcassette kan een speelveld worden gekozen met behulp van de F1 toets. Er zij 64 verschillende spelsituaties in negen moeilijkheidsgraden, waarin JACK (de speler dus) op ballonnen moet jagen. Hij kan daarbij gebruik maken van blokken, die verplaatsbaar zijn, stapelbaar en kunnen drijven op het vijvertje onderaan op het scherm. Kortom, niet iets echt nieuws, maar uiteindelijk toch iets voor de Plus/4 en de C-16.
De eerste indruk van de spelhandleiding is gunstig als je de twaalf vertalingen ziet. Er is zelfs een Arabische bij ! Dat verandert als je de Nederlandse (nou ja, Nederlandse) gebruiksaanwijzing leest. Die Nederlandse versie doet wat houterig aan. Alsof iemand hem met een Nederlands woordenboek in de hand heeft opgesteld. Zonder overigens gehinderd te zijn door enige kennis van de Nederlandse taal. Erg slordig, maar we zien het vaker. Ik zelf zeg liever joystick dan "stuurstok"!

Adventure

Gunstiger doen de drie adventures aan. De naam Scott Adams zal bij velen een belletje doen rinkelen. In de Adams-traditie zijn deze tekstadventures echte hersenbrekers. Alleen tekst mag dan saai lijken, PIRATE ISLAND, STRANGE ODYSSEY en ATOMIC MISSION zijn alles behalve slaapverwekkend. De pittige speelplots staan borg voor uren adventureplezier. Al moet wel worden vermeld, dat de avonturier in spe wel de Engelse taal redelijk moet beheersen. De

HELP-toets van de computer heeft in het spel een duidelijke functie. Soms krijgt de avonturier hulp, een andere keer moet hij het zelf maar uitzoeken.

PIRATE ISLAND

Het avontuur begint, zoals dat vaak het geval is, op een geheel onverwachte plaats. In dit geval in een flat in Londen. Daar zijn ook de aanwijzingen voor het avontuur te vinden. Een met bloed geschreven aanwijzing van ene Long John Silver in een half vergaan boek en een geheime gang achter een boekenkast nodigen iedere avonturier uit op zoek te gaan naar de verborgen schatten op Pirate Island.
Dat het "slechts" tekst is, zoals een verwende avonturier zou kunnen stellen, doet aan de kwaliteit van dit adventure niets af. Het is eigenlijk net als het lezen van een boek: je kunt je eigen beeld vormen van het verhaal. Bij een film is dat beeld al voor je gevormd.

STRANGE ODYSSEY

Inderdaad een zeer vreemd ruimte avontuur. Een motor, die niet wil starten, een console met een rode knop en een klem zittend service-paneel vormen de inleiding van een reis vol avontuur, spanning en loerende gevaren. Dodelijke methaangassen en buitenaardse wezens bedreigen de avonturier.

ATOMIC MISSION

Heel anders van karakter is het derde avontuur. Geen ruimtewezens of piraten. Wel een grote nucleaire reactor. Mensen werken er niet, want de reactor is geheel geautomatiseerd. Toch loopt er iemand rond: een saboteur. De avonturier moet hem vinden, maar er is niet veel tijd. De bom, in zijn lichaam geïmplanteerd, staat nog op veilig. Maar voor hoelang?
Eigenlijk een heel beklemmend adventure. Want waar ligt de scheiding tussen fictie en werkelijkheid. Of is het misschien een toekomstvisie van Scott Adams. Er moet gezegd worden, dat de hele situatie in het avontuur iet ver naast de werkelijkheid staat. Een spannend avontuur en vooral een die te denken geeft. Jereinste Sience-Fiction.

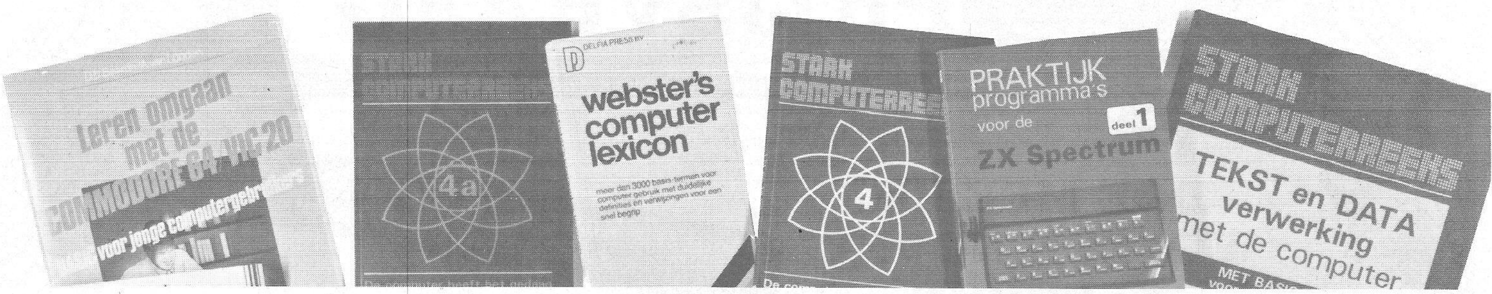
Tot zover de Scott Adams Adventures. Volgens de verpakkingen zijn er slechts drie (deze drie dus) van deze adventures op de markt. Het leuke van Adams is de consequent aangehouden opbouw van de verhalen. De commando's zijn in alle drie gelijk, zodat je, als je er een hebt gespeeld, een ander zo kan beginnen.

Plus/4 LOGO

De laatste programmacassette in deze bespreking is de LOGO-module voor de Plus/4 computer van Commodore. Niet geschikt voor de C-16 helaas, zoals we abusievelijk vermelden, maar dat zijn we wel gewend van Commodore! We denken overigens, dat het wel werkt met de uitbreidingsmodule van Artic, maar we hebben (nog) geen zekerheid. Bij de LOGO-module zelf is een zeer uitgebreid handboek en een utility-disk. Op deze disk staan verschillende procedures en demo's voor direct gebruik.
Deze LOGO-versie komt, evenals die voor de C-64, van Terrapin (MIT) en is van goede kwaliteit. Het is natuurlijk logisch, dat Commodore dezelfde, zij het aangepaste versie, gebruikt voor de Plus/4. Alle LOGO primitieven zijn aanwezig en ook de animatie- en geluidsmogelijkheden ontbreken niet. Ook hier zijn de LOGO primitieven in het Engels, maar dat is nauwelijks een bezwaar te noemen, ook niet voor Nederlandse kinderen. Als je meer te weten wil komen over de Commodore LOGO versies, raad ik je aan de reeds verschenen Commodore-Info's nog eens door te bladeren en we komen binnenkort uitgebreid op Logo terug. Toch is het jammer, dat de LOGO alleen voor de Plus/4 computer beschikbaar is. Een C-16 versie zou m.i. veel beter in de markt liggen, gezien het aantal verkochte machines in Nederland. Maar ja, eigenlijk worden de twee "nieuwe" computers van Commodore niet echt serieus genomen. Maar we doen ons best, Commodore-Info heeft zijn eigen "minderheden-beleid".

Bob Munniksma ●

Boeken ◇ Boeken ◇ Boeken

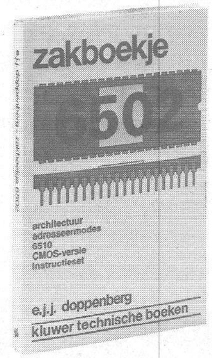


Peter Kremer geeft kritisch commentaar op een aantal boeken.

Zakboekje 6502

E.Doppenberg, uitgave KTT *f* 21,-

Machinetaal-liefhebbers kunnen niet buiten dit soort werkjes, waar alle informatie over hun favoriete processor in is opgeslagen. Dit boekje levert alles, wat men nodig heeft, maar echt lekker doorlezen is er niet bij, dit is een naslagboekje, om er bij te hebben liggen om snel de juiste gegevens op te zoeken. Het heeft een handig formaat en geeft een overzicht van 6502 en afgeleide processoren zoals de 6510 en de 65C02, de CMOS variant.



Het Groot, Compleet, Nederlandstalig Commodore 64 Boek

P. Neumann : uitgave Kockelkoren Publ., *f* 45,-

Als je dit boek koopt moet je eerst goed kijken of het wel goed in elkaar zit. Mij is het overkomen dat bij een exemplaar 10 pagina's ontbraken en andere pagina's niet in de juiste volgorde zaten. Mijn tweede exemplaar was beter, doch ook hier zaten meerdere pagina's door elkaar. Op de omslag staat vermeld: "voor beginners en gevorderden". Mijn mening is dat de beginner hier weinig aan heeft, tenzij hij alleen maar en zonder er iets van te begrijpen, programma's wil overtypen. De schrijver legt bij de meeste programma's de werking niet uit. Dat moet je zelf maar uitzoeken schrijft hij. Door de slechte drukkwaliteit zijn verschillende programma's moeilijk te ontcijferen en vergt het heel wat puzzelen en experimenteren voordat de zaak loopt. Bovendien is de herkomst van een aantal programma's "dubieus" en daarmee is dit boek ook al niet onze favoriet.

The Elementary Commodore 64

William B. Sanders: uitgave Datamost, *f* 37,50
ISBN 0-8359-1702-9

Dit Engelstalige boek is werkelijk een van de betere boeken voor de beginner. Vlot en bovenal humoristisch geschreven en voorzien van talrijke cartoons. Op een zeer heldere manier worden alle mogelijkheden van de 64 uiteen gezet. Daarnaast wordt er uitgebreid ingegaan op het programmeren in BASIC. Echt de moeite waard.

Tips & Trucs voor de Commodore 64

Angerhausen, Englisch, en Gerrits: Uitgave Data Becker/Bruna, *f* 59,-.
ISBN 90-229-3322-9

Een nuttig boek voor hen die uit de beginnersschoenen zijn. Wat de titel beloofd wordt volledig waar gemaakt. Alle programma's worden uitvoerig uitgelegd. Van de machine-taal programma's is er ook altijd een BASIC laadprogramma afgedrukt. Er wordt zeer uitgebreid ingegaan op de werking van CP/M. Degene die niet met dit besturingssysteem werkt moet dat maar voor lief nemen.

Mastering the Commodore 64

door Mark Greenshields: Uitgave van Interface London, *f* 40,-.
ISBN 0-907563-38-4

Met deze titel als vlag wordt de lading niet gedekt. Veel belangrijke zaken worden zeer summier afgedaan (zoals bijv. geluid en grafische mogelijkheden, en rand apparatuur). Er staat een leuk stukje in over machine-taal, maar iemand die daar wat over wil weten is beter uit met een boekje dat alleen dat onderwerp behandelt.
Jammer, maar helaas is dit Engelstalige werkje niet aan te bevelen.

Doe-het-zelf programma's op de C-64

D.Kreutner, uitgave Academic Service *f* 39,50

Een behoorlijk stevige stapel programma's en de doe-het-zelver kan er wel even tegenaan, als hij ze allemaal wil overtikken. Het gaat om relatief ingewikkelde programma's, die ook wel praktijkgericht zijn. Zo is er een aantal boekhoudtoepassingen, prijslijst, chequeadministratie en puntenlijst. Maar ook veel spelletjes, database programma's, utilities en een EDIT functie om gegevensbestanden te genereren. In totaal 40 programma's, die bepaald niet erg simpel zijn, men is uitgegaan van de volledige benutting van de Basic mogelijkheden. Prettig voor wie ook echt iets verwacht van zijn moeizaam overtikken. Jammer, dat een regeltester ontbreekt, maar gezien de goede leesbaarheid van de listings geen onoverkomelijk bezwaar.

Basic Min

Een rubriek van Nico Baaijens me

Het strand of zwembad roept natuurlijk zeer dwingend nu de zomer dan toch nog eindelijk gekomen is. Voor de korte avonden bestaat er natuurlijk geen betere manier om van uw computer te genieten dan met een miniatuurtje, namelijk leuk en toch niet te lang. Gelukkig konden we weer een keuze maken uit een ruime hoeveelheid materiaal ingestuurd door Commodore-Info-lezers.

Niet alles is altijd even briljant maar er zitten toch weer diverse huzarenstukjes bij, die even ingeklopt moeten worden.

Het begint een goede gewoonte te worden onze rubriek voor snelle intoetsers te beginnen met de Tien met de Griffel. Die gaat deze keer naar Wim Tubbax uit Laakdal-Veerle, België.

Cryptografie

Het miniatuurtje van Wim Tubbax bevat een fantastisch algoritme (computer-verwerkingsmethode) om teksten om te zetten in geheimschrift en andersom om geheim geschreven teksten weer terug te vertalen naar klare taal. Het programmaatje is meer dan alleen een aardigheidje. Een nuttige toepassing ontstaat als de algoritme wordt verwerkt in een Basic-programma, waarmee teksten worden versluierd. Zou de diskette of de tape in verkeerde handen komen, dan kan de dief, cq. de oneerlijke vinder geen misbruik maken van de informatie, tenzij hij bereid is een jaartje of wat puzzelend door te brengen om de code te kraken.

Nog een toepassing is het versluieren van DATA-statements in een programma. Als dat programma bijvoorbeeld een woorden- raadspelletje is, zal degene, die de listing intypt, er geen plezier meer aan beleven als hij de oplossingen al heeft overgetypt.

```
1 A$="ABCDEFGHIJKLMNOPQR
  STUVWXYZ1234567890 ": D$="": INPUT"GEEF
  TEKST";C$
2 B$="QW1ER2TY3UI4OP5AS6DF7GH8JK9LZ X
  XCVBMN0": INPUT"<Shift/Clr>1 = CODEREN 2 =
  DECODEREN";X
3 IFX=2THEN 6
4 FORI=1TOLEN(C$): FOR N=1TO37: IF MID$(A$,N
  ,1)=MID$(C$,I,1) THEN D$=D$+MID$(B$,N,1)
5 NEXTN: NEXTI: PRINT D$: END
6 FORI=1TOLEN(C$): FOR N=1TO37: IF MID$(B$,N,1)=
  MID$(C$,I,1) THEN D$=D$+MID$(A$,N,1)
7 NEXTN: NEXTI: PRINT D$: END
```

Belangrijk: als dit programma correct is ingetypt, moet na de prompt: "Geef tekst" de volgende code worden ingevoerd: P3150WQQ3URPD en vervolgens de decodeer-optie (2) worden gekozen.

Blokjes

Miniatuurtjes die, al of niet in kleur en met geluid, de raarste dingen met het beeldscherm uithalen, blijven binnenkomen. Ik moet ze steeds vaker terzijde leggen omdat ze in herhalingen gaan vervallen. De inzending van Harmen Siemelink uit Ede was echter wel aardig:

```
10 PRINT "<Shift/Clr>": I=RND(1)*159: POKE 53280,0:
   POKE 53281,0
20 FORT=1024TO2023STEP1024/I: POKE T,160: POKE
   54272 +T,I: I=I+1
30 IFI=>160THEN I=1
40 NEXT: GOTO20
```

Nog meer grafische grappen van dezelfde auteur:

```
10 POKE 53280,0: POKE 53281,0: INPUT"<Shift/Clr>
   GETAL";A
20 FORT=0TOA: PRINT "(PUR)/(OMLAAG)";: NEXT
30 FORT=0TOA: PRINT "(BLU)/(OMHOOG)";: NEXT
40 GOTO20
```

Bij dit miniatuurtje is het boven de 22 niet zo leuk meer. Voor speciale effecten gelden de getallen: 0,1,3,4,7,9,13,18,19 en 20.

Raadselachtig is de 'schuifband':

```
10 : : : : : : : : : : : : : : : : : : : POKE 53281,6:POKE
   53281,14: GOTO10
```

Sommige inzenders zijn te bescheiden. Ze hebben geen hoge pet op van zichzelf, zoals B. Martensen uit Ermelo. Hij schrijft: "Mijn bijdrage is niet erg interessant, maar de rubriek moet toch vol". Nou ja, zo erg is het niet hoor! Maar vooruit.

```
10 TI$="000000": PRINT "<Shift/Clr>"
20 PRINT "<HOME>"
30 PRINT TI$
40 IFTI$="000005"THEN POKE 53280,8: POKE 53281,1
50 IFTI$="000010"THEN POKE 53280,1: POKE 53281,8
60 IFTI$="000015"THEN POKE 53280,7: POKE 53281,0
70 IFTI$="000020"THEN POKE 53280,0: POKE 53281,7
80 IFTI$="000025"THEN 100
90 GOTO20
100 END
```

We blijven nog even bij de klok en maken hem tikkend volgens de methode van H. Link uit Amsterdam.

```
10 TI$="130000": PRINT "<Shift/Clr>"
20 PRINT "<HOME><2×NEER>"
   PRINT LEFT$(TI$,2)": "MID$(TI$,3,2)": "RIGHT$(TI$,2)
30 POKE 54296,15: POKE 54277,8: POKE 54276,65: FOR
   T=1TO840: NEXT: POKE 54276,0: GOTO20
```

iatuurtjes OO O

korte tot zeer korte programma's.

De eerste inzendingen voor onze programma-prijsvraag, waarbij het gaat om iets langere (maar ook niet te lange) programma's, komen inmiddels ook binnen. Denk eraan, daarmee valt een Novex kleurenmonitor te winnen. De sluitingsdatum voor inzending is 25 september 1985, U heeft dus nog even de tijd. De voorwaarden van onze wedstrijd staan elders in dit nummer nog een keer afgedrukt.

"Wat er allemaal gebeurt op het scherm, weet ik niet, maar het is wel mooi", schrijft Roland Overwijk uit Alphen a/d Rijn. Hier is zijn miniatuurtje:

```
1 FORA=1TO255: POKE 53272,A NEXT: GOTO1
```

Deze inzender vond ook nog een aardige POKE voor de C64:

```
POKE 808,100
```

DataPuls

Omdat het stomvervelend is grote blokken met DATA-regels van een of andere leuke game te moeten naspeuren op fouten als er geen checksum-controle is, heeft Robin Kramer uit Amsterdam een 'knutseltje' geschreven dat als subroutine enorm van pas zal komen. In plaats van moeizaam naspeuren en vergelijken tussen listing en wat er op het scherm staat, ga je nu onderuit gezakt in je stoel zitten en laat de VIC, de CBM of welk Commodore-mirakel dan ook, het werk doen door de DATA bliepend langs te lopen. Robin noemt het een 'DataPuls'.

```
60000 RESTORE: POKE 36878,15: SD=36876:
      SN=36874: HS=128: TS=158: ES=188
60001 FORI=1TO<aantal data>: READD$: FQ=LEN(D$)
60002 ONFQGOTO 60003,60006,60008: POKE SN,254:
      GOSUB60011: POKE SN,0: GOTO60010
60003 EH=VAL(D$)
60004 GOSUB60013: ONFGOTO60010
60005 FORJ=1 TO EH: POKE SD,ES: GOSUB60011:
      POKE SD, 0: GOSUB60012: NEXT: GOTO 60010
60006 TT=VAL(LEFT$(D$,1)): EH=VAL(MID$(D$,2,1))
60007 FORJ=1TOTT: POKE SD,TS: GOSUB 60011:
      POKE SD, 0: GOSUB 60012: NEXT: GOTO60004
60008 HT=VAL(LEFT$(D$,1)): TT=VAL(MID$(D$,2,1)):
      EH=VAL(MID$(D$,3,1))
60009 FOR J=1 TO HT: POKE SD,HS: GOSUB60011:
      POKE SD,0 : GOSUB 60012: NEXT: GOTO60007
60010 GOSUB60012: NEXTI: POKE 36878,0: END
60011 FORT=1TO200: NEXT: RETURN
60012 FORT=1TO50: NEXT: RETURN
60013 IF EH=0 THEN POKE SN,175: GOSUB60011:
      POKE SN,0 : GOSUB60012: F=1: RETURN
60014 F=0: RETURN
```

Het is geen simpel programma. Enige nadere toelichting is dus op z'n plaats. De foutzoek-routine is bedoeld voor kleine getallen, zoals POKE 's. Het is ook geen geschikte manier om fouten te zoeken in DATA van rekenkundige programma's want dat zou te veel verschillende bloepjes en bliepjes geven en dat is niet meer te volgen.

Robin Kramer stelt voor een demo te doen door een DATA-regel in te voeren met 500,50,5,0 erin. Je hoort dan vijfmaal honderdtal, tiental, eenheid en één nul.

Zoals het programma hierboven staat, is het geschikt voor de VIC20. Commodore-64 gebruikers moeten in regel 60000 in de variabele SD het adres voor de hoogste toongenerator plaatsen (SD = 54273) en in SN de waarde voor de laagste toongenerator (SN = 54272). Ook moet de poke in 60000 anders 36878→54296 en de pokes voor het geluid worden 54277,64 54276,17 54278,128. Nog een opmerking over regel 60001, waar de FOR/NEXT-loop moet lopen tot het aantal te controleren DATA-statements. Als dit teveel is om op te tellen, kan een te hoog getal worden gegeven. Het programma loopt er dan voorbij met de bekende OUT OF DATA-foutmelding.

Beeldbeweging

Uit Ferwerd in Friesland kwam van M.R. Themmen een miniatuurtje, dat het beeldscherm in zijn geheel een rare beweging laat maken.

```
10 FORA=0TO15: POKE 53270,A
20 FORB=0TO80: NEXT: NEXT
30 FORA=15TO0STEP-1: POKE 53270,A
40 FOR B=0TO80: NEXT: NEXT
```

Iets dergelijks, maar op een heel andere wijze geprogrammeerd, werd ingezonden door M. Hammie uit Dordrecht, die als suggestie geeft zijn routine te gebruiken bij de explosie in een game.

```
10 FOR T=192 TO 199
20 POKE 53270,T: POKE 53265,T-176: NEXT
30 FOR T=199 TO 192 STEP-1
40 POKE 53270,T: POKE 53276,T-176: NEXT
50 GOTO 10
```

Voor een 'geinig geluidje' verschafte Stephan Zwanikken uit Hattem dit Basic-regeltje:

```
5 FORQ=54296TO54295STEP-.1: FORT=0TO15:POKE
  Q,T: NEXT: NEXT: GOTO5
```

En om het beeldscherm de aandacht van een spelletjesspeler te laten trekken:

```
5 PRINT "<Shift/Clr>": FORQ=1TO65: POKE 53344,Q:
  POKE 53345,Q: NEXT
```

Tekenen met de cursor, die met de Commodore- en de CTRL-toetsen van kleur verandert, doe je met het miniatuurtje van Steven van der Velde uit Vlissingen.

```
10 PRINT "<Shift/Clr>"
20 GETA$
30 PRINT        A$;"<RVS/ON><SPATIE><RVS/OFF>
  <CRSR/LEFT>";
40 GOTO20
```

Het 'inkleuren' van de cursor kan ook worden geprogrammeerd met:

```
30 PRINT A$;"<RVS/ON><KLEUR><SPATIE> <RVS/
  OFF><CRSR/LEFT>";
```

Van Leo Groenendijk uit Delft kreeg ik een aardig programmaatje dat een eindloos mozaïek laat zien.

```
10 POKE 53280,0: POKE 53281,0: PRINT "<Shift/Clr>"
20 B=1024: A=RND(1): POKE B+ A*B,250: POKE B*
  54+A*B,2+ RND(1)*14: GOTO20
```

Volgens L.J. van der Goes uit Born in Limburg laat het onderstaande miniatuurtje alle karakters van de C-64 voorbij trekken op het scherm.

```
10 PRINT "<Shift/Clr>"
20 FORL=0TO191: A=INT(L*RND(1)+1)
30 PRINT CHR$(A);: NEXT
40 GOTO20
```

Ik had nog meer ingezonden miniatuurtjes willen plaatsen, maar enkele inzendingen geven problemen. Ze lopen niet goed en waarschijnlijk zijn er type- of overschrijf-fouten in geslopen. Daarom vraag ik de volgende inzenders hun huiswerk nog eens over te doen: Marcel Jansen uit Noordwijkerhout en M. de Big uit Zevenhuizen. Erik Hendrikx uit Westervoort vraag ik zijn bijdragen wat beter toe te lichten. Ik begrijp echt niet wat de bedoeling is van zijn miniatuurtjes.
Ad Struijk uit Gouda zond een interessante brief, maar de listing ontbreekt of is ergens zoek geraakt. De programmaatjes van Mark Westbroek over het omzetten van getallen naar andere talstelsels kan ik helaas niet plaatsen, omdat dit al zo vaak aan de orde is geweest. Ook voor het maken van de tafels van J. Poort uit Waalwijk had ik graag iets anders. Volgende keer beter dus.

Ook de volgende Commodore/Info moet weer een miniatuurtjes-rubriek hebben, maar dat hangt helemaal af van de actievelingen en vindingrijken onder de lezers, die ook in deze (tot dusver) wat aarzelende zomer de computer niet vergeten. Aarzel dus niet en houd uw kleine, handige, leuke en leerzame miniatuurprogrammaatjes niet alleen voor uzelf. ●

Print-out

Met o.a. Kokos I en II - Letterraam - Grafiek - Doolhof - C-16 Maestro - Vic-invaders - Vingervlug - Dubbelborder (versie 2) - Sprites opnieuw - Datamaker - Delete - Smooth scroll etc.

In de listingsrubriek van dit zomernummer, tref je naast de inmiddels traditionele kleine listings van leuke truukjes en handige utilities voor het programmeren in Basic, ook een paar langere listings aan. We hebben deze lappen machinetaal opzettelijk bewaard tot de zomervakantie, omdat de meesten van ons nu iets ruimer in hun vrije tijd zitten. Voor de regenachtige zomerdagen ligt er nu dus het nodige typewerk te wachten.

Naast het superspel KOKOS hebben we nog een ander machinetaalprogramma, nl. het surfspel surfmania van Antoine vd. Berg. Maar we willen u niet aandoen, nogmaal 32 KB aan code te moeten intypen. Toch is SURFMANIA een erg goed en origineel spel, echt de moeite waard. Daarom hebben we voor onze speciale Zomercassette van INFOLIST, waar dus de listings uit no. 5 en 6 op staan, het er als EXTRA programma maar bijgezet. Het is dus deze keer extra aantrekkelijk om zo'n cassette a ƒ 22,50 te bestellen via storting op giro 3157656 tnv. Infolist met vermelding Zomercassette.

Besluit je zelf alles in te typen, dan wens ik je sterkte en succes bij het nakijken van de data-regels. Verder nog een opmerking over de listings in het algemeen. Bovenaan de listings staat vermeld voor welk type Commodore-computer het programma is bestemd. Ook zul je hier en daar tussen aanhalingstekens hele rijen hoofdletters aantreffen. Deze moeten exact worden overgetypt, met de computer in "TEXT"-mode (kleine letters). We hebben dit bewust gedaan, omdat de grafische tekens, die uiteindelijk op je scherm zullen verschijnen nogal eens voor verwarring zorgen, omdat ze zoveel op elkaar lijken. Vandaar.

Als laatste nog een opmerking over de listings voor VIC, C-16 en PLUS/4. We krijgen relatief weinig goede listings toegestuurd in deze categorie. Daarom een oproep aan alle programmeurs van die machines. Stuur je VIC en C-16 programma's voor publikatie aan ons op, en wie weet, worden ze geplaatst, wat weer voor een leuke beloning zorgt. Bovendien help je de andere gebruikers op deze manier.

Jan Bodzinga.



... IK BEGRIJP NIET WAT JIJ IN KOKOS ZIET, JOOP...

Kokos

Een aap, drie palmen, een sloot, kookpot en talloze kokosnoten, dat zijn de ingredienten voor dit spel, dat geschreven werd door Harrie van Lammeren uit Vogelenzang.

In de eerste plaats mijn kompliment voor dit prachtige werkstuk. Het gebeurt niet elke dag, dat we zo'n listing op ons bureau krijgen. Want het spel is werkelijk af. Goed geschreven en leuk om te spelen. Je aap moet zoveel mogelijk noten vangen, wassen en koken, zonder zelf door een kokosnoot te worden verpletterd. Daarnaast moet hij ook nog strijd leveren tegen de computerklok. Het geheel bestaat uit twee delen, kokos-1 en -2, waarbij eerst deel 1 moet worden geladen en gerund. Daarna laad je deel 2 en daarmee kan de pret beginnen.

Ik hoef verder niets te zeggen over de spelregels, nadat je het een keer hebt gespeeld wijst alles zich vanzelf.

Hoewel dit een erg lang stuk vervelend typewerk oplevert hebben we gemeend, gezien de kwaliteit van de listing, voor de lezers toch dit programma te moeten publiceren. Sterkte bij het intoetsen.

```
 1 rem kokos 1 / commodore-64
 2 rem door harrie van lammeren
 3 rem vogelenzang   02502-5576
 4 rem
10 rem   eerst dit programma laden en
20 rem   runnen. daarna kokos 2 laden
30 rem   en runnen.
40 rem
42 print"32 seconden geduld a.u.b."
43 dimm(5555):clr
45 poke35000,170
```

Kokos I (vervolg)

```
 50 reada:ifa>=0thenpoke33792+b,a:b=b+1
    :goto50
 60 reada:ifa>=0thenpoke34304+c,a:c=c+1
    :goto60
 65 reada:ifa>=0thenpoke34560+d,a:d=d+1
    :goto65
100 data173,28,212,73,255,141,15,212,20
    1,255,208,5,169,128,141,18,212
110 data174,6,207,208,3,76,51,133,173,0
    ,208,201,40,240,3,76,254,132
115 data173,5,207,240,10,169,182,141,24
    8,119,169,183,141,249,119,169,,141,
    5,207
120 data173,224,206,208,3,76,232,132,17
    3,,220,41,16,205,225,206,208,3,76,2
    31
130 data132,173,225,206,73,16,141,225,2
    06,240,77,138,10,168,169,210,153,1,
    208
140 data169,192,133,251,169,92,133,252,
    162,11,160,2
150 data189,226,206,145,251,202,136,16,
    247,165,251,24,105,8,133,251,138,16
    ,235
160 data162,182,142,248,119,232,142,249
    ,119,174,6,207
170 data206,224,206,208,19,169,0,141,4,
    207,169,13,157,39,208
171 data173,27,208,61,40,207,141,27,208
180 data234,96,138,10,168,169,222,153,1
    ,208
190 data169,192,133,251,169,92,133,252,
    162,11,160,2
200 data189,238,206,145,251,202,136,16,
    247,165,251,24,105,8,133,251,138,16
    ,235
210 data162,186,142,248,119,232,142,249
    ,119,169,0,141,15,212,169,16,141,18
    ,212
220 data169,22,141,19,212,169,17,141,18
    ,212,96,189,39,208,201,253,240,68
230 data173,27,208,29,30,207,141,27,208
    ,169,6,141,224,206,96
240 data173,2,208,201,29,208,46,173,0,2
    20,41,16,208,39
250 data189,39,208,201,253,208,32
260 data142,49,207,169,113,157,222,207,
    169,255,157,228,207
270 data169,0,141,6,207,173,27,208,29,3
    0,207,141,27,208,169,1,141,4,207
280 data173,1,207,201,4,208,10,206,251,
    206,208,8,169,2,141,251,206,32,0,12
    0
281 data96,-1
290 data174,49,207,240,63,32,0,126,138,
    10,168,173,250,206,24,105,22,141,25
    0,206
300 data185,,208,105,,153,,208,185,,208
    ,201,56,144,33,185,1,208,201,204,14
    4,26
310 data169,0,141,49,207,157,246,207,16
    9,0,141,15,212,169,40,141,19,212
320 data169,17,141,18,212,32,160,124,96
    ,-1
330 data120,169,255,162,255,157,0,93,20
    2
335 data208,250,169,0,133,251,133,253,1
    69,205,133,252,169,207,133,254,162,
    2
340 data160,0,177,251,145,253,136,208,2
    49,198,252,198,254,202,208,240
350 data173,30,208,141,29,207,162,180,1
    60,2,136,208,253,202,208,248
352 data32,,124,32,,129,32,,132,32,,134
    ,32,,136,32,,137,173,252,206,208,21
    7
354 data162,5,189,248,207,208,210,202,1
    6,248
356 data88,96,-1
360 fork=27648to28030:reada:pokek,a:next
370 fork=49152to49378:reada:pokek,a:next
380 fork=49408to49457:reada:pokek,a:next
390 reada:ifa>=0thenpoke38912+e,a:e=e+1
    :goto390
395 reada:ifa>=0thenpoke34816+f,a:f=f+1
    :goto395
396 reada:ifa>=0thenpoke35072+g,a:g=g+1
    :goto396
400 print"laad nu het tweede programma
    en run het."
1000 data,2,,,138,,,40,,2,8,128,2,138,12
     8,,170,128,,170,,,40,,,235,,,253,,3
     ,255
1010 data192,3,253,64,3,255,192,3,253,64
     ,3,255,192,3,253,64,3,255,192,,253,
     ,
1020 data255,,,253,,,60,,
1030 data,,,,,,,,,,,,,,,,,,,,,2,8,,,2,,,
     160,,2,128,136,,136,8,,193,40,32,19
     3,4
1040 data32,193,4,32,193,16,8,209,16,15,
     241,16,3,213,64,,253,,,63,,
1050 data,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
     ,,,,,,,,,,,,8,,8,,32,3,16,64,,243,6
     5,
1060 data17,,131,,40,32,205,16,
1070 data,240,63,3,92,255,3,119,215,3,12
     7,171,3,95,173,,245,85,,7,85,,5,213
     ,,1
1080 data127,,1,85,,,85,,,5,,,19,,3,255,
     ,3,255,,,15,,,15,,,3,,,12,,,48,,,25
     2,
1090 data,,63,,,255,,,95,,,239,,1,165,,5
     ,85,,21,85,,85,95,,3,245,12,21,85,2
     52,1
1100 data85,255,192,21,3,252,,,63,255,,3
     ,255,,,3,,,3,,,3,,,,,,3,,,63,
1110 data243,192,,253,112,,247,112,,255,
     112,,125,192,,87,,,212,,,84,,,84,,,
     84,,
1120 data80,,,240,,,240,,,240,,,240,,,24
     0,,,255,,,240,240,,204,12,
1125 data3,3,192,15,,63
1130 data3,255,252,12,,,,,,54,27,252,255
     ,3,,,,,55,34,195,255,63,3,53,41
1140 data177,251,145,253,136,208,249,96
1145 data162,4,74,110,6,192,74,46,7,192,
     14,6,192,46,7,192,202,208,239
1147 data173,7,192,162,0,129,251,96
1150 data169,255,133,251,162,108,134,252
     ,232,134,254
1160 data74,133,253,160,128,32,26,192
1170 data169,48,141,246,109,141,249,109,
     169,240,141,252,109
1180 data162,,142,187,109,162,3,142,190,
     109
1190 data169,255,133,251,162,108,134,252
     ,232,134,254
1200 data133,253,160,,32,26,192
1210 data160,25,190,,192,136,185,,192,13
     3,251
1220 data136,185,,192,157,,110,157,128,1
     10,232,232,232,228,251,208,239
1230 data136,16,227
1240 data169,,133,251,169,192,133,253,16
     9,111,133,252,169,108,133,254
1245 data162,21,138,72
1250 data160,2,177,253,32,34,192,230,251
     ,208,2,230,252,136,16,242
1260 data160,2,230,253,208,2,230,254,136
     ,16,247
1265 data104,170,202,208,222,230,251,208
     ,2,230,252,230,253,208,2,230,254
1270 data238,5,192,173,5,192,201,9,208,1
     98,96
1271 data114,148,119,152,119,19,152,119,
     232,119,120,,116,152,119,,148,219,1
     52
```

Kokos I (vervolg)

```
1272 data219,,108,219,112,219,,68,219,72
     ,219,,28,219,32,219,5,152,219,232,2
     19
1273 data9,240,216,152,219,5,,216,240,21
     6
1280 data80,144,84,148,84,149,245,157,9,
     10,41,42,41,170,175,186
1290 data170,,,3,15,,,,,31,255,255,252,,
     1,3,31,251,155,53,117,233,200,128,
1300 data170,255,127,63,31,15,7,1,,254,2
     54
1310 data32,16,8,4,2,3,1,,255,127,63,31,
     15,135,227,252,254,254
1320 data30,15,15,7,7,3,1,1,63,31,143,22
     7,252,255,255,255,254,254
1330 data128,130,65,97,48,48,56,28,,,4,1
     31,225,240,124,127,4,6,3,3,193,241,
     120
1335 data30,59,25,29,15,143,247,251,127
1340 data170,170,170,1,5,69,69,67,35,51,
     51
1350 data132,2,142,12,192,72,169,3,141,1
     0,192,104
1360 data72,17,253,145,253,177,251,141,1
     1,192,104,72
1365 data162,8,74,46,6,192,202,208,249,1
     73,6,192,129,251,13,11,192,145
1370 data251,152,24,105,64,168,104,206,1
     0,192,208,214,164,2,174,12,192,96
1380 data169,192,133,251,169,92,133,252
1390 data169,255,162,,129,251,230,251,20
     8,2,230,252
1400 data164,251,192,64,208,242,164,252,
     192,95,208,236
1410 data169,144,133,251,169,32,133,253,
     169,64,133,254,133,252,162,,169,6,7
     2
1420 data169,4,72,160,7,189,16,152,201,1
     70,240,14,201,254,240,15,32,131,152
1430 data232,136,16,238,24,144,16,169,,2
     4,144,2,169,255
1440 data160,7,32,131,152,136,16,250,232
1450 data165,253,24,105,16,133,253,165,2
     54,105,,133,254
1460 data165,251,56,233,16,133,251,165,2
     52,233,,133,252,104,56,233,1,208,18
     4
1470 data165,253,24,105,0,133,253,165,25
     4,105,1
1480 data133,254,165,251,24,105,128,133,
     251,165,252,105,1,133,252
1490 data104,56,233,1,208,149
1500 data169,96,133,251,169,71,133,252,1
     60,0,177,251,240,34,162,8
1510 data74,144,7,102,253,102,254,56,176
     ,5
1520 data102,253,102,254,24,102,253,102,
     254,202,208,234
1525 data165,253,145,251,160,8,165,254,1
     45,251
1527 data198,251
1530 data165,251,201,255,208,208,198,252
     ,165,252,201,63,208,200
1540 data169,,133,251,169,2,133,252,169,
     88,133,253,169,69,133,254,169,19,72
     ,162
```

```
1550 data7,160,128,189,8,152,17,253,145,
     253,138,72,152,9,8,168,189,0,152,23
     4
1560 data234,234,17,253,145,253,104,170,
     152,56,233,72,168,16,224,230,253,20
     8,2
1565 data230,254,202,16,213
1570 data24,165,253,105,56,133,253,165,2
     54,105,1,133,254,104,56,233,1,208,1
     91
1580 data96,160,49,162,3,185,,193,149,25
     1,136,202,16,247,162,0
1590 data185,,193,129,251,230,251,208,2,
     230,252
1600 data165,251,197,253,208,239,165,252
     ,197,254,208,233
1610 data136,16,217,169,45,141,30,119,14
     1,70,119,169,18,141,109,119,141,110
     ,119
1615 data96,-1
1620 data173,200,92,208,31,173,6,207,208
     ,27
1622 data173,49,207,208,21,173,95,93,41,
     1,208,14,160,72,169,255,153,63,93,1
     52
1623 data56,233,8,168,16,244,96
1625 data206,19,207,208,250,169,110
1630 data141,19,207
1640 data172,255,206,174,253,206,185,10,
     207,29,0,93,157,0,93
1650 data152,73,3,168,174,254,206,185,10
     ,207,29,0,93,157,0,93
1660 data206,255,206,16,73,169,3,141,255
     ,206,173,253,206,24,105,8,141,253,2
     06
1670 data173,254,206,56,233,8,141,254,20
     6,201,95,208,46,173,6,207,141,220,2
     06
1675 data169,0,141,224,206
1680 data169,0,141,6,207,169,1,141,4,207
     ,169,0,141,15,212,169,128,141,18,21
     2
1690 data169,192,141,19,212,169,129,141,
     18,212,169,255,141,219,206,96,-1
1700 data169,114,133,254,173,220,206,208
     ,1,96,24,106,106,106,133,253
1710 data144,2,230,254,162,3,138,72,173,
     27,212,41,63,168,173,27,212,41,3,17
     0
1720 data177,253,61,20,207,145,253,104,1
     70,202,208,230
1730 data206,219,206,208,20,174,220,206,
     173,21,208,61,40,207,141,21,208
1740 data169,0,157,246,207,141,220,206,9
     6,-1

*** EINDE LISTING ***
```

```
kokos 1  checksum

regel 1   178     regel 150  7      regel 330  180
regel 2   180     regel 160  62     regel 335  109
regel 3   204     regel 170  145    regel 340  88
regel 4   143     regel 171  126    regel 350  222
regel 10  144     regel 180  77     regel 352  63
regel 20  87      regel 190  209    regel 354  124
regel 30  38      regel 200  10     regel 356  24
regel 40  143     regel 210  7      regel 360  113
regel 42  168     regel 220  236    regel 370  125
regel 43  206     regel 230  148    regel 380  127
regel 45  83      regel 240  89     regel 390  178
regel 50  115     regel 250  95     regel 395  185
regel 60  109     regel 260  63     regel 396  184
regel 65  121     regel 270  162    regel 400  110
regel 100  65     regel 280  72     regel 1000  169
regel 110  212    regel 281  124    regel 1010  103
regel 115  65     regel 290  33     regel 1020  127
regel 120  140    regel 300  232    regel 1030  50
regel 130  204    regel 310  167    regel 1040  146
regel 140  209    regel 320  11     regel 1050  168
```

```
regel 1060 167   regel 1290 202   regel 1560  205
regel 1070 184   regel 1300 218   regel 1565  244
regel 1080 40    regel 1310 169   regel 1570  151
regel 1090 205   regel 1320 68    regel 1580  194
regel 1100 180   regel 1330 194   regel 1590  209
regel 1110 191   regel 1335 83    regel 1600  177
regel 1120 18    regel 1340 232   regel 1610  2
regel 1125 48    regel 1350 41    regel 1615  124
regel 1130 73    regel 1360 207   regel 1620  18
regel 1140 100   regel 1365 81    regel 1622  177
regel 1145 31    regel 1370 214   regel 1623  213
regel 1147 145   regel 1380 93    regel 1625  137
regel 1150 211   regel 1390 151   regel 1630  116
regel 1160 238   regel 1400 69    regel 1640  95
regel 1170 57    regel 1410 125   regel 1650  146
regel 1180 16    regel 1420 41    regel 1660  158
regel 1190 211   regel 1430 103   regel 1670  158
regel 1200 188   regel 1440 114   regel 1675  201
regel 1210 114   regel 1450 92    regel 1680  96
regel 1220 65    regel 1460 101   regel 1690  183
regel 1230 119   regel 1470 204   regel 1700  220
regel 1240 33    regel 1480 67    regel 1710  98
regel 1245 8     regel 1490 96    regel 1720  38
regel 1250 65    regel 1500 89    regel 1730  98
regel 1260 205   regel 1510 227   regel 1740  64
regel 1265 134   regel 1520 74
regel 1270 140   regel 1525 175
regel 1271 154   regel 1527 233   ready.
regel 1272 166   regel 1530 238
regel 1273 168   regel 1540 38
regel 1280 1     regel 1550 136
```

```
  1 rem  kokos 2
  2 rem
  3 rem  eerst kokos 1 laden en runnen.
  4 rem  daarna dit programma laden en
  5 rem  runnen.
  6 rem
  9 ifpeek(35000)<>170thenprint"laad en
     run eerst het andere programma.":end
 10 printchr$(14)"[CLR-HOME][oranje]":poke
    53281,1:poke53280,9
 20 poke52,160:poke56,160
100 a$(0,2)="[RVS-aan]▀ ▀[RVS-uit] ▪[RVS-aan
    ] [RVS-uit]▪":a$(0,3)="[RVS-aan][2xspat
    ie][RVS-uit]▪[RVS-aan][2xspatie][RVS-ui
    t]▀":a$(0,4)="[RVS-aan][4xspatie][RVS-u
    it]▀
110 a$(0,5)="[RVS-aan][4xspatie][RVS-uit]
    ▪ ":a$(0,6)="[RVS-aan][2xspatie][RVS-ui
    t]▀[RVS-aan][2xspatie][RVS-uit]▪":a
    $(0,7)="[RVS-aan]▪ ▪[RVS-uit] ▀[RVS-aan]
     [RVS-uit]▀
120 a$(1,2)=" ▬[RVS-aan] ▀[RVS-uit]▪ ":
    a$(1,3)="[RVS-aan]▌[4xspatie][RVS-uit]
     ":a$(1,4)="[RVS-aan][2xspatie][RVS-uit
    ]▀[RVS-aan]▪ [RVS-uit]▐
130 a$(1,5)="[RVS-aan][2xspatie][RVS-uit]
    ▪[RVS-aan]▀ [RVS-uit]▐":a$(1,6)="[RVS-a
    an]▌[4xspatie][RVS-uit] ":a$(1,7)="
     [RVS-aan]▬ ▪[RVS-uit]▀ "
140 a$(4,2)="▪[RVS-aan][3xspatie][RVS-uit]
    ▬ ":a$(4,3)="[RVS-aan][5xspatie][RVS-ui
    t]▐":a$(4,4)="▀[RVS-aan] [RVS-uit]▬
    ▪[RVS-aan]▬[RVS-uit] "
150 a$(4,5)="▪▪[RVS-aan]▬▪ ▀[RVS-uit] ":
    a$(4,6)="[RVS-aan][5xspatie][RVS-uit]
    ▌":a$(4,7)="▀[RVS-aan]▪[2xspatie]▪[RVS-
    uit] "
160 fork=2to7:a$(2,k)=a$(0,k):a$(3,k)=a
    $(1,k):next:b$="[6xspatie]"
170 a$="[links]":fork=0to4:a$="[6xrechts]"
    +a$:a$(k,0)=b$:a$(k,1)=b$:forl=7to0
    step-1
180 print"[HOME]";:form=1to7:printa$;a$
    (k,m):nextm,l,k
200 print"[2xneer] [RVS-aan]In dit spel
     ben je een aapje, dat een "
```

Kokos II (vervolg)

```
210 print" [RVS-aan]aantal wonderlijke
     bomen in zijn ter- "
220 print" [RVS-aan]ritorium heeft staa
     n. De kokosnoten[3xspatie]"
230 print" [RVS-aan]van deze bomen zijn
      pas eetbaar, als[2xspatie]"
240 print" [RVS-aan]ze naar beneden val
     len. Omdat dit al- "
250 print" [RVS-aan]leen vandaag gebeur
     t, moet je zo veel "
260 print" [RVS-aan]mogelijk kokosnoten
      verzamelen. De ko-
270 print" [RVS-aan]kosnoten moeten gew
     assen en gekookt[3xspatie]"
280 print" [RVS-aan]worden voordat ze v
     ergaan. Iedere ko- "
290 print" [RVS-aan]kosnoot, die gewass
     en in de pan zit,[2xspatie]"
300 print" [RVS-aan]vergaat niet meer e
     n is 1 punt waard. "
310 print" [RVS-aan]Je kunt bewegen met
      de stuurknuppel en
320 print" [RVS-aan]met de vuurknop.[22xspa
     tie]"
330 print"[neer][7xspatie][RVS-aan]36 s
     econden geduld A.U.B.";:poke52,64:poke
     56,64
395 sys49213:sys39103
397 reada:ifa>=0thenpokei+35328,a:i=i+1
     :goto397
400 fork=23744to23771:reada:pokek,a:next
     :fork=24072to24083:reada:pokek,a:next
410 b=13:fork=22765to23422:reada:pokek,
     a:b=b+1:ifb=32thenb=0:k=k+288
420 next
422 reada:ifa>=0thenpoke30720+c,a:c=c+1
     :goto422
424 reada:ifa>=0thenpoke31744+d,a:d=d+1
     :goto424
426 reada:ifa>=0thenpoke31884+e,a:e=e+1
     :goto426
428 reada:ifa>=0thenpoke32256+f,a:f=f+1
     :goto428
429 reada:ifa>=0thenpoke32512+g,a:g=g+1
     :goto429
430 reada:ifa>=0thenpoke33024+h,a:h=h+1
     :goto430
432 fork=52437to52693:reada:pokek,a:next
     :fork=52694to52735:pokek,0:next
433 poke52734,255
589 print"[op]"chr$(141)"[37xspatie]";
590 print"[op]"chr$(141)"[RVS-aan][rechts]
     Druk op de vuurknop.[20xlinks][RVS-
     uit]";:b=0
591 fork=0to17:printchr$(148);:c=peek(5
     6320)and16
592 ifc=16thenb=1
593 if(c=0)and(b=1)then599
594 next:print"[18xspatie]";:fork=0to17
     :printchr$(20);
595 c=peek(56320)and16:ifc=16thenb=1
596 if(c=0)and(b=1)then599
598 next:goto591
599 poke53265,peek(53265)and239:sys3940
     8
600 poke56578,peek(56578)or3:poke56576,
     (peek(56576)and252)or2:poke53272,20
     8
610 poke53265,peek(53265)or32:poke53270
     ,peek(53270)or16
620 poke53280,1
660 v=53248:pokev+21,3:pokev+23,3:pokev
     +28,255:pokev+37,8:pokev+38,11
670 pokev+39,1:pokev+40,1:pokev+27,0
680 pokev,4:pokev+2,28:pokev+16,3:pokev
     +1,192:pokev+3,192:sys35341
690 s=54272:fork=0to24:pokes+k,0:next:poke
     s+24,15:pokes+1,200:pokes+6,0
695 pokes+14,255:pokes+15,255:pokes+18,
     128:pokes+8,50:pokes+12,5
700 poke17784,255
900 poke53265,peek(53265)or16:poke16544
     ,0:sys34560:pokev+21,0
901 poke53265,peek(53265)and239
910 poke53270,peek(53270)and239:poke532
     65,peek(53265)and223:poke53272,21
920 poke56576,(peek(56576)and252)or3:poke
     56578,peek(56578)and252
925 r=0:p=0:q=0:k=52445:l=53213:n=100
930 form=0to2:p=p+peek(k+m)*n:q=q+peek(
     l+m)*n:n=n/10:next
940 ifq>=pthenform=0to2:pokek+m,peek(l+
     m):next
945 fork=0to23:pokes+k,0:next:pokes+5,9
     :pokes+12,10:pokes+19,12
946 pokes+4,17:pokes+11,17:pokes+18,17:
     sys35328
950 print:printchr$(142)"[CLR-HOME]"tab(
     15)"hiscore ";
951 fork=52445to52447:ifpeek(k)>0thenr=
     1
960 ifr=1thenprintchr$(peek(k)+48);
962 next:print:print"[neer]"tab(13)"[oranje
     ]▁▬[RVS-aan]¯¯¯ [RVS-uit]▌[RVS-aan]
     ▌ [RVS-uit]▌[2xrechts][RVS-uit]▬▁
963 printtab(10)"▬[RVS-aan]¯[rechts][6xspat
     ie][RVS-uit]▌[RVS-aan]▌[2xspatie][recht
     s][RVS-uit]▪[RVS-aan][2xspatie][rechts]
     ¯[RVS-uit]▬
964 printtab(9)"[RVS-aan]▒[2xspatie][rechts
     ][2xspatie][RVS-uit]¯¯[RVS-aan]▌ [RVS-u
     it]▌[RVS-aan]▌[2xspatie][RVS-uit]▌[RVS-
     aan][3xspatie][rechts][2xspatie]▒
965 printtab(8)"[RVS-aan]▒[3xspatie][rechts
     ][2xspatie][2xrechts][RVS-aan]▌ [RVS-ui
     t]▌[RVS-aan]▌ [RVS-uit]▌[RVS-aan] ▪
     [2xspatie][rechts][2xspatie]▬▒
966 printtab(7)"[RVS-aan]▒ [RVS-uit]▒ [RVS-
     aan]▬[rechts][2xspatie]¯¯[2xspatie]
     [RVS-uit]▌[RVS-aan]▌ [RVS-uit]▌[RVS-aan
     ]▪[RVS-uit]▪[RVS-aan][2xspatie][rechts]
     [2xspatie][RVS-uit]▬▪
967 print"[GROEN][3xspatie][RVS-aan]▒[RVS-u
     it]▒ [oranje][RVS-aan]▒ [RVS-uit]▒[rech
     ts][RVS-uit]▁▬ [RVS-aan][6xspatie][RVS-
     uit]▌[RVS-aan]▌ [RVS-uit]▌[RVS-aan]
     ▌[rechts][2xspatie][rechts][4xspatie]
     [RVS-uit]▬
968 print"[GROEN] [RVS-aan]▒[rechts] [recht
     s][oranje][RVS-uit]▪[RVS-aan] [RVS-uit]
     ▒[rechts][RVS-aan]▬[2xspatie][rechts]
     [bruin][2xspatie][RVS-uit]¯¯[RVS-aan]
     ▌ [RVS-uit]▌[RVS-aan]▌ [RVS-uit]▌[2xspa
     tie][RVS-aan][2xspatie][oranje][rechts]
     [2xspatie][rechts][RVS-uit]¯[RVS-aan]
     ▬▁[RVS-uit]▪
969 print"[GROEN] ▒[RVS-aan]▪ ▒[oranje]
     | [RVS-uit]▬[RVS-aan][bruin]¯[3xspatie]
     [oranje][rechts][2xspatie][2xrechts]
     ▌ [RVS-uit]▌[RVS-aan]▌ [RVS-uit]▌[2xspa
     tie][RVS-aan][2xspatie][bruin][rechts]
     [3xspatie]¯[RVS-uit][oranje]▬▁
970 print"[2xspatie][GROEN][RVS-aan]▬[2xspa
     tie][oranje][7xspatie][rechts][bruin]
     [2xspatie][2xrechts]▌ [RVS-uit]▌[RVS-aa
     n]▌ [RVS-uit]▌[2xrechts][RVS-aan][2xspa
     tie][rechts][oranje][7xspatie]":print
     " [GROEN]▪[RVS-aan][3xspatie][RVS-uit]
     ▌
971 print"[2xspatie][RVS-aan]▒[2xspatie]
     [bruin][7xspatie][rechts][oranje][2xspa
     tie][2xrechts]▌ [RVS-uit]▌[RVS-aan]
     ▌[6xspatie][rechts][bruin][7xspatie]"
972 print" [RVS-aan][GROEN]▒[2xspatie][RVS-
     uit]▒[RVS-aan][bruin]| ▬[oranje][4xspat
     ie][rechts][bruin][2xspatie][RVS-uit]
     | [RVS-aan]▌ [RVS-uit]▪[RVS-aan]▌[6xspa
     tie][rechts][oranje][2xspatie][bruin]
     [RVS-uit]¯¯ [RVS-aan] |
```

```
973 print" [GROEN]▒[RVS-aan]▒ [rechts][brui
    n][RVS-uit]▘[RVS-aan] ▒[2xrechts][2xspa
    tie][rechts][oranje]| [RVS-uit]▌ [RVS-a
    an]| [RVS-uit]▌[RVS-aan]▌ [RVS-uit]
    ▌[5xspatie][RVS-aan][bruin][2xspatie]
    [RVS-uit]▄[RVS-aan]‾[2xspatie][RVS-uit]
    ▪
974 print" [GROEN]|[RVS-aan] [RVS-uit]▒
    [RVS-aan]▒[rechts][RVS-uit][bruin]▒
    [RVS-aan] ▒[rechts][2xspatie][rechts]
    ▌ [RVS-uit]▌[rechts][RVS-aan][2xspatie]
    [RVS-uit]▌[RVS-aan]▌ ▘[RVS-uit]▄▄[RVS-a
    an]‾‾[rechts][4xspatie]▄
975 print"[2xspatie][GROEN][RVS-aan]▪[RVS-u
    it]▪[3xspatie][bruin]▒[RVS-aan] ▒[2xspa
    tie][rechts]▌ ▌[RVS-uit]|[RVS-aan][2xsp
    atie][RVS-uit]|[RVS-aan]▌[6xspatie]
    [rechts][3xspatie]▒
976 printtab(8)"▒[RVS-aan][3xspatie][rechts
    ][RVS-uit]▌[RVS-aan] |[RVS-uit]▌[RVS-aa
    n][2xspatie][rechts]▌ ▪▄[RVS-uit]‾‾
    ‾ [RVS-aan][2xspatie][RVS-uit]▒▒
977 printtab(9)"▒[RVS-aan][2xspatie][rechts
    ][RVS-uit]|[RVS-aan][2xspatie]▌ |[recht
    s]▌ [RVS-uit]▌_▄▄ [RVS-aan][2xspatie]"
978 printtab(10)"‾[RVS-aan]_[2xrechts][4xsp
    atie]▌[rechts]▌[6xspatie][rechts]_[RVS-
    uit]‾
979 printtab(14)"‾[RVS-aan]▄__[RVS-uit]
    ▌ [RVS-aan]▌ _▄[RVS-uit]‾‾
980 printtab(16)"[neer]score ";
981 r=0:fork=53213to53215:ifpeek(k)>0then
    r=1
982 ifr=1thenprintchr$(peek(k)+48);
983 next:print:print"[neer]"tab(10)"dru
    k op de vuurknop.";:poke53265,peek(
    53265)or16
984 b=0
985 c=peek(56320)and16:ifc=16thenb=1
986 if(c=0)and(b=1)then599
999 goto985
1000 data120,169,26,141,20,3,169,138,141
     ,21,3,88,96,120,169,49,141,20,3,169
     ,234
1005 data141,21,3,88,96,162,2,160,14
1010 data254,213,206,189,213,206,153,1,2
     12,221,216,206,208,15,169,0,157,213
     ,206
1020 data169,16,153,4,212,169,17,153,4,2
     12,202,152,56,233,7,168,16,219
1030 data76,49,234,-1
1620 data192,240,252,254,255,255,255,255
     ,,,136,170,170,170,234,234,,,136,17
     0
1625 data170,170,171,171
1630 data3,15,191,191,234,250,250,254,25
     5,255,255,255,171,175,175,191
1640 data63,255,255,,,,,,254,254,254,,,,
     ,,192,240,240,,,,,,,3,3,255,255,255
     ,255
1650 data255,255,63,63,254,254,254,254,2
     54,254,254,254,240,240,240,240,240,
     240
1660 data204,204,3,3,3,3,3,3,3,,63,63,63
     ,63,,,,,255,255,255,255,,,,
1670 data204,204,204,204,204,204,12
1675 data138,72,173,220,207,133,251,173,
     255,207,133,252,162,3,134,2
1680 data173,27,212,41,127,168,173,27,21
     2,41,3,170,173,1,207,201,2,176,17
1690 data189,20,207,57,64,113,153,64,113
     ,166,2,202,208,221
1695 data24,144,54,173,1,207,201,4,240,4
     7,234,234,234,234
1700 data189,10,207,49,251,25,64,113,153
     ,64,113,166,2,202,208,191,173,1,207
1710 data201,2,208,20,173,0,207,208,15,1
     69,128,141,4,212,169,208,141,5,212
1720 data169,129,141,4,212,173,1,207,201
     ,4,240,38
1730 data238,0,207,208,3,238,1,207,173,1
     ,207,201,4,240,3,76,11,122
1740 data169,128,141,4,212,169,200,141,1
     ,212,169,0,141,5,212,169,0,141,6,21
     2
1750 data173,0,220,41,12,201,12,208,43
1760 data173,2,207,240,19,162,182,173,4,
     207,208,2,162,186
1770 data142,248,119,232,142,249,119,76,
     11,122
1780 data162,191,173,4,207,208,2,162,195
     ,142,249,119,232,142,248,119,76,11,
     122
1790 data173,0,220,41,8,240,3,76,103,121
     ,169,0,141,2,207,173,16,208,41,3,24
     0,7
1800 data173,2,208,201,29,240,208,162,18
     9,173,4,207,208,2,162,193
1810 data206,3,207,208,13,169,4,141,3,20
     7,173,5,207,73,1,141,5,207
1820 data173,5,207,208,2,232,232,142,249
     ,119,232,142,248,119
1830 data173,5,207,240,17,173,3,207,201,
     1,208,10,169,128,141,4,212,169,129
1840 data141,4,212
1850 data238,0,208,208,8,173,16,208,9,1,
     141,16,208
1860 data238,2,208,208,8,173,16,208,9,2,
     141,16,208
1870 data173,6,207,168,10,170,240,27
1880 data24,173,2,208,105,12,157,0,208,1
     73,16,208,41,2,208,2,144,9,173,16,2
     08
1885 data25,30,207,141,16,208
1890 data76,11,122
1900 data169,1,141,2,207,173,16,208,41,3
     ,208,10,173,,208,201,40,208,3
1905 data76,165,120
1910 data162,180,173,4,207,208,2,162,184
     ,206,3,207,208,13,169,4,141,3,207
1920 data173,5,207,73,1,141,5,207,173,5,
     207,208,2,232,232
1930 data142,248,119,232,142,249,119,173
     ,5,207,240,17,173,3,207,201,1,208,10
1940 data169,128,141,4,212,169,129,141,4
     ,212
```

Kokos II (vervolg)

```
1950 data206,2,208,173,2,208,201,255,208
     ,8,173,16,208,41,253,141,16,208
1960 data206,0,208,173,0,208,201,255,208
     ,8,173,16,208,41,254,141,16,208
1970 data173,6,207,168,10,170,208,3,76,1
     1,122,56,173,,208,233,12,157,0,208
1975 data173,16,208,41,1,240,5,144,3,76,
     11,122
1980 data173,16,208,57,40,207,141,16,208
     ,234,234,104,170
1990 data96,-1
2000 data173,200,92,208,42,206,7,207,208
     ,37
2010 data173,194,92,73,2,141,194,92
2020 data173,202,92,73,170,141,202,92
2030 data173,210,92,73,170,141,210,92
2040 data173,218,92,73,128,141,218,92
2050 data169,40,141,7,207
2060 data206,8,207,208,68,169,27,141,8,2
     07,162,130,173,9,207,240,2,162,172
2070 data169,103,133,253,169,92,133,254,
     134,2,162,5,189,13,207,168,138,72
2080 data166,2,189,,207,232,134,2,145,25
     3,104,170,136,152,221,23,207,208,235
2090 data169,91,133,254,202,208,224,206,
     9,207,16,5,169,1,141,9,207,96,-1
2100 data254,220,207,189,220,207,201,10,
     208,9,169,0,157,220,207
2110 data202,32,140,124,96,162,3,32,140,
     124,169,207,133,252,169,93,133,254
2120 data189,220,207,10,10,10,24,105,50,
     133,251,138,10,10,10,24,105,224,133
     ,253
2130 data160,7,177,251,145,253,136,16,24
     9,232,138,201,4,208,221,96,-1
2140 data169,,133,251,169,108,133,252,16
     9,114,133,254
2150 data254,222,207,208,3,254,228,207,1
     38,10,168,24,189,234,207,125,222,20
     7
2160 data157,234,207,185,1,208,141,48,20
     7,125,228,207,153,1,208,205,48,207,
     240
2170 data48,189,240,207,240,43,138,24,10
     6,106,106,133,253,144,2,230,254
2180 data189,240,207,56,233,3,157,240,20
     7,168,200,200
2190 data177,251,145,253,136,152,221,240
     ,207,16,245
2200 data173,27,208,61,40,207,141,27,208
     ,96,-1
2205 data138,10,168
2210 data56,185,,208,237,,208,72,173,2,2
     07,240,7,104
2220 data201,24,16,64,48,92,104,201,1,48
     ,57,16,85,172,6,207,208,51,169,3
2230 data157,246,207,142,6,207
2231 data169,63,141,253,206,169,135,141,
     254,206,169,3,141,255,206
2232 data138,10,168,169,210,153,1,208
2233 data160,72,169,0,153,63,93,152,56,2
     33,8,168,16,244
2234 data169,,141,4,207,96
2235 data169,177,153,248,119,169,2,153,2
     46,207,169,1,141,4,207,169,0,141,6,
     207
2236 data152,10,168,169,213,153,1,208,96
2240 data72,173,2,207,240,7,104,201,255,
     16,27,48,5,104,201,26,48,20
2243 data72,173,2,207,240,7,104,201,239,
     16,142,48,224,104,201,42,48,135,16,
     217
2250 data172,6,207,240,18,169,1,153,246,
     207,169,,141,6,207,152,10,168
2260 data169,213,153,1,208,169,1,141,4,2
     07,169,,141,,207
2270 data169,0,141,1,207,169,128,141,4,2
     12,169,,141,6,212
2280 data169,10,141,1,212,169,12,141,5,2
     12,169,129,141,4,212,169,,133,252
2285 data173,2,207,73,1,168
2290 data185,248,119,160,6,10,38,252,136
     ,208,250,133,251,141,220,207,165,25
     2,24
2300 data105,64,133,252,141,255,207,160,
     127,177,251,153,64,113,136,16,248
2301 data172,2,207,169,198,153,248,119,1
     52,73,1,168,169,197,153,248,119,96,
     -1
2310 data206,39,207,208,15,169,255,141,3
     9,207,173,38,207
2320 data201,20,240,3,238,38,207,162,7,3
     2,31,129,202,224,1,208,248,96
2330 data189,246,207,208,39,173,252,206,
     56,240,35,173,27,212,205,38,207,176
     ,27
2340 data169,1,157,246,207,169,5,157,39,
     208
2350 data169,,157,222,207,157,228,207,16
     9,61,157,234,207
2360 data208,4,208,122,176,120,206,254,2
     07,208,3,206,252,206,169,114,133,25
     4
2370 data138,24,106,106,106,133,253,144,
     2,230,254,169,0
2380 data160,62,145,253,136,16,251
2390 data138,24,105,200,157,248,119,138,
     72,10,133,2,168,169,61,153,1,208
2400 data173,27,212,201,84,144,249,153,0
     ,208
2401 data162,14,228,2,240,22,189,1,208,2
     01,90,176,15,56,189,,208,249,,208
2402 data201,12,16,4,201,245,16,218,202,
     202,224,2,208,224
2409 data104,170,173,16,208,61,40,207,14
     1,16,208
2410 data169,63,157,240,207,173,21,208,2
     9,30,207,141,21,208,96
2420 data189,246,207,201,1,208,90,32,0,1
     26,173,1,207,201,4,208,28
2430 data138,10,168,185,1,208,201,100,14
     4,26,173,29,207,61,30,207
2440 data240,3,32,0,127,189,246,207,201,
     3,240,8
2450 data173,29,207,61,30,207,208,18
2460 data138,10,168,185,1,208,201,213,14
     4,34,234,234,234,234,234,234,234,23
     4
2470 data169,2,157,246,207
2480 data169,177,157,248,119,169,30,157,
     212,207,169,128,141,11,212
2490 data169,129,141,11,212,96
2500 data189,246,207,201,2,208,34,222,21
     2,207,208,29
2501 data169,30,157,212,207,189,248,119
2510 data201,178,208,14,173,21,208,61,40
     ,207,141,21,208,169,,157,246,207
2520 data254,248,119,96,-1
2980 data,,,45,90,180,255,
2990 data,,,,16,3,15,191,,,136,,,136,192
     ,240,252,35,175,191,34,170,170,34,1
     70
2995 data170,226,250,254,,2,4,,,
3000 data,4,1,4,1,,,20,5,2,3,12,48,192
3010 data,8,16,24,88,110,252,243,207,63,
     255,4,12,23,56,
3020 data1,2,4,8,16,32,64,128,,255,254,2
     53,251,247,239,223,191,127,,
3030 data255,207,51,51,51,51,51,207,255,
     207,15,207,207,207,207,3
3040 data255,207,51,243,195,15,63,3,255,
     207,51,243,207,243,51,207
3050 data255,51,51,51,3,243,243,243,255,
     3,63,15,243,243,243,15
3060 data255,207,51,63,15,51,51,207,255,
     3,243,243,207,207,207,207
3070 data255,207,51,51,207,51,51,207,255
     ,207,51,51,195,243,51,207
3080 data195,195,3,3,163,3,3,3,235,239,2
     52,118,86,88,152,238,238,254,47,167
     ,9
```

```
3090 data37,137,10,12,12,12,12,12,12,12,
     12,131,239,239,255,255,43,15,63,63,
     12
3100 data195,195,163,131,3,131,163,35,23
     5,238,221,85,86,102,104,104,238,250
3110 data173,149,37,37,165,137,12,12,12,
     14,12,12,12,14,11,187,251,250,254,9
     ,14
3120 data62,62,14

*** EINDE LISTING ***


kokos 2  checksum
regel 1    72     regel 397  195    regel 910  232
regel 2    143    regel 400  20     regel 920  226
regel 3    197    regel 410  68     regel 925  73
regel 4    180    regel 420  130    regel 930  229
regel 5    147    regel 422  157    regel 940  13
regel 6    143    regel 424  169    regel 945  213
regel 9    81     regel 426  179    regel 946  200
regel 10   215    regel 428  178    regel 950  176
regel 20   192    regel 429  177    regel 951  12
regel 100  64     regel 430  171    regel 960  199
regel 110  83     regel 432  23     regel 962  19
regel 120  106    regel 433  100    regel 963  249
regel 130  136    regel 589  155    regel 964  94
regel 140  107    regel 590  148    regel 965  249
regel 150  106    regel 591  213    regel 966  52
regel 160  47     regel 592  179    regel 967  146
regel 170  178    regel 593  116    regel 968  219
regel 180  241    regel 594  80     regel 969  157
regel 200  183    regel 595  11     regel 970  196
regel 210  57     regel 596  116    regel 971  62
regel 220  143    regel 598  228    regel 972  190
regel 230  140    regel 599  13     regel 973  182
regel 240  33     regel 600  22     regel 974  41
regel 250  216    regel 610  30     regel 975  15
regel 260  33     regel 620  246    regel 976  25
regel 270  44     regel 660  162    regel 977  75
regel 280  163    regel 670  136    regel 978  37
regel 290  153    regel 680  235    regel 979  161
regel 300  150    regel 690  15     regel 980  216
regel 310  223    regel 695  228    regel 981  110
regel 320  22     regel 700  106    regel 982  199
regel 330  42     regel 900  146    regel 983  112
regel 395  121    regel 901  45     regel 984  36
```

```
regel 985   11    regel 1960  106   regel 2320  7
regel 986   116   regel 1970  252   regel 2330  232
regel 999   47    regel 1975  148   regel 2340  42
regel 1000  4     regel 1980  142   regel 2350  167
regel 1005  181   regel 1990  124   regel 2360  103
regel 1010  244   regel 2000  19    regel 2370  86
regel 1020  74    regel 2010  150   regel 2380  87
regel 1030  216   regel 2020  232   regel 2390  168
regel 1620  99    regel 2030  230   regel 2400  66
regel 1625  105   regel 2040  249   regel 2401  218
regel 1630  38    regel 2050  157   regel 2402  164
regel 1640  187   regel 2060  17    regel 2409  255
regel 1650  213   regel 2070  236   regel 2410  193
regel 1660  116   regel 2080  96    regel 2420  63
regel 1670  114   regel 2090  169   regel 2430  67
regel 1675  169   regel 2100  227   regel 2440  202
regel 1680  203   regel 2110  7     regel 2450  188
regel 1690  241   regel 2120  205   regel 2460  101
regel 1695  184   regel 2130  18    regel 2470  215
regel 1700  55    regel 2140  7     regel 2480  150
regel 1710  244   regel 2150  115   regel 2490  151
regel 1720  47    regel 2160  209   regel 2500  205
regel 1730  98    regel 2170  112   regel 2501  96
regel 1740  80    regel 2180  253   regel 2510  162
regel 1750  1     regel 2190  158   regel 2520  212
regel 1760  197   regel 2200  163   regel 2980  190
regel 1770  171   regel 2205  119   regel 2990  115
regel 1780  215   regel 2210  201   regel 2995  217
regel 1790  171   regel 2220  169   regel 3000  240
regel 1800  81    regel 2230  151   regel 3010  85
regel 1810  111   regel 2231  91    regel 3020  215
regel 1820  90    regel 2232  234   regel 3030  21
regel 1830  254   regel 2233  112   regel 3040  81
regel 1840  58    regel 2234  209   regel 3050  178
regel 1850  105   regel 2235  225   regel 3060  75
regel 1860  108   regel 2236  132   regel 3070  78
regel 1870  184   regel 2240  158   regel 3080  194
regel 1880  141   regel 2243  191   regel 3090  124
regel 1885  89    regel 2250  19    regel 3100  62
regel 1890  63    regel 2260  136   regel 3110  189
regel 1900  245   regel 2270  182   regel 3120  16
regel 1905  119   regel 2280  198
regel 1910  207   regel 2285  255
regel 1920  182   regel 2290  252   ready.
regel 1930  96    regel 2300  221
regel 1940  68    regel 2301  198
regel 1950  109   regel 2310  166
```

Letterraam

Een programma van Rob Danen, dat -in Basic- van alle in te typen karakters een vergroot beeld (dots) weergeeft op het scherm, waardoor je de opbouw van het betreffende symbool kunt bekijken. Bovendien is eerst de karaktergenerator naar RAM gekopieerd (regel 150), zodat ook de karakteropbouw te veranderen is. De RAM adressen van het symbool worden ook op het scherm gezet. Leuk om te bekijken en mee te experimenteren.

```
 1 rem letterraam / commodore-64
 2 rem door rob danen
 3 rem den haag / 070-252436
 4 rem
10 poke53280,246:poke53281,246:print"[WIT]
20 rem bescherm character-set
30 rem ======================
40 print"[CLR-HOME][2xneer]even geduld"
   :print
50 print"vergroter door:"
60 print"EEEEEEEEEEEEEE"
70 print"UCCCCCCCCI"
80 print"Brob danenB"
90 print"JCCCCCCCCK"
100 rem copieer characters
110 rem ==================
120 printchr$(142):poke52,48:poke56,48:
    clr
130 poke56334,peek(56334)and254
140 poke1,peek(1)and251
150 fori=0to1104:pokei+12288,peek(i+532
    48):next
160 poke1,peek(1)or4
170 poke56334,peek(56334)or1
180 :
190 rem start vergroten
200 rem ===============
210 print"[CLR-HOME]":print"toets symbo
    ol"
220 geta$:ifa$=""then220
230 print"[CLR-HOME][HOME]";a$:p1=peek(
    1024):p=peek(1024):p=p*8:a=12288+p:
    a1=53248+p:print"[CLR-HOME]"
240 forx=0to7:a(x+1)=peek(a+x):nextx
250 print"UCCCCCCCCI"
260 rem reken dec > bin uit
270 rem ==================
280 forc=1to8
290 forx=1to8
300 b=int(a(c)/2):b=b*2:b=a(c)-b
310 ifb=0thena$(x)=a$(x)+" "
320 ifb=1thena$(x)=a$(x)+"Q"
330 a(c)=int(a(c)/2)
340 nextx
350 rem print char.
360 rem ==========
370 print"B";:fori=8to1step-1:printa$(i
    );:a$(i)="":nexti:print"B";c
380 nextc
390 print"JCCCCCCCCK":print"[neer] 1234
    5678"
400 print"[neer]symbool..............:
     ";chr$(34);a$;chr$(34)
410 print"ascii code............:";asc( a$)
420 print"plaats in char-set [1]:";a1
430 print"plaats in char-set [2]:";a
440 print"[1] :[RVS-aan]originele[RVS-uit]
    [3xspatie]character-set"
450 print"[2] :[RVS-aan]gecopieerde
    [RVS-uit] character-set"
460 print"[neer]druk een toets"
470 geta$:ifa$=""then470
480 goto190
```

regel 1	235	regel 230	36
regel 2	12	regel 240	177
regel 3	160	regel 250	147
regel 4	143	regel 260	217
regel 10	25	regel 270	22
regel 20	126	regel 280	131
regel 30	205	regel 290	152
regel 40	72	regel 300	236
regel 50	11	regel 310	18
regel 60	104	regel 320	228
regel 70	86	regel 330	65
regel 80	170	regel 340	218
regel 90	77	regel 350	104
regel 100	118	regel 360	46
regel 110	217	regel 370	186
regel 120	194	regel 380	197
regel 130	42	regel 390	86
regel 140	127	regel 400	19
regel 150	211	regel 410	122
regel 160	28	regel 420	64
regel 170	193	regel 430	16
regel 180	58	regel 440	232
regel 190	217	regel 450	103
regel 200	34	regel 460	139
regel 210	59	regel 470	104
regel 220	97	regel 480	35

Grafiek

Gerrit Koffeman maakte dit staaltje van grafiekopbouw. Kort en goed te begrijpen. Na het ingeven van de nodige getallen krijg je een kleurrijke staafgrafiek op je scherm. Denk erom dat je de in te geven waarden niet te ver uitelkaar laat lopen. Dit programma leent zich prima om te worden ingebouwd in je eigen software, want inplaats van het intypen van de variabelen, kun je ze natuurlijk door de computer laten genereren of uitrekenen.

```
  1 rem grafiek / commodore-64
  2 rem door gerrit koffeman
  3 rem uit urk
  4 rem
 10 rem grafiek
 20 print"[CLR-HOME][WIT]":dimhe(36):a1
    $="[ROOD][GEEL]"
 30 gosub210
 40 poke53280,0:poke53281,0:gosub310
 50 forp=0toh:ifhe(p)=0then150
 60 if(p/2)-int(p/2)=0thenq=0:goto80
 70 q=1
 80 ss$=mid$(a1$,1+q,1)
 90 print"[HOME][4xrechts]
    [22xneer][rechts]";
100 gosub180
110 print"[RVS-aan]"ss$" [RVS-uit]";
120 i=1:ifhe(p)=1then150
130 print"[op][links][RVS-aan]"ss$"
    [RVS-uit]";
140 i=i+1:ifi<he(p)then130
150 nextp
160 geta$:ifa$=""then160
170 run
180 ifp=0then200
190 fori=1top:print"[rechts]";:nexti
200 return
210 input"naam van het bestand";n$
220 print:input"hoeveel gegevens [max.
    35]";h
```

```
230 h=h-1:fori=0toh
240 print"gegeven"i+1;:inputhe(i):nexti
250 mx=0:forj=0toh:ifhe(j)>mxthenmx=he(
    j):nextj
260 ifmx=0thenrun
270 sc=22/mx:forj=0toh:he(j)=he(j)*sc
280 ifhe(j)-int(he(j))=>.5thenhe(j)=int
    (he(j))+1:goto300
290 he(j)=int(he(j))
300 nextj:print"[CLR-HOME]":return
310 forx=0to21:printtab(4)"[WIT]P":next
320 print"[4xrechts]";
330 forx=0to35:print"[WIT]P";:next
340 sp=int((40-len(n$))/2)
350 print"[HOME]"spc(sp)"[RVS-aan][WIT]"
    n$"[RVS-uit]"
360 print"[HOME][neer]";
370 forx=22to1step-1:printleft$(str$((m
    x/22)*x),4):next:return
```

```
 grafiek

                             regel 170    138
 regel 1     243             regel 180    246
 regel 2     215             regel 190    21
 regel 3     115             regel 200    142
 regel 4     143             regel 210    90
 regel 10     136            regel 220    175
 regel 20     252            regel 230    240
 regel 30     32             regel 240    40
 regel 40     128            regel 250    201
 regel 50     177            regel 260    67
 regel 60     210            regel 270    79
 regel 70     52             regel 280    50
 regel 80     226            regel 290    8
 regel 90     50             regel 300    62
 regel 100     38            regel 310    106
 regel 110     202           regel 320    140
 regel 120     63            regel 330    170
 regel 130     248           regel 340    32
 regel 140     73            regel 350    5
 regel 150     210           regel 360    60
 regel 160     100           regel 370    57
```

Doolhof

Kort maar krachtig. De computer maakt op het scherm z'n eigen doolhof, zet zichzelf er ergens middenin en gaat vervolgens op zoek naar de uitgang. Je kunt er zeker van zijn, dat vroeg of laat de uitgang wordt gevonden, als de C264 tenminste zichzelf niet heeft ingesloten.

Dit is nu een voorbeeld van een (korte) listing, waarbij je alleen maar de schermlokaties (1024 voor C-64) hoeft te veranderen om het te kunnen gebruiken op een VIC of C-16.

```
  1 rem doolhof-64  / commodore-64
  2 rem door michiel & jeroen soede
  3 rem hoensbroek  045-218448
  4 rem
 10 print"[CLR-HOME]"
 20 fort=40to959:ifint(rnd(1)*3.3)=1then
    poke1024+t,160
 30 nextt
 40 fort=1024to1063:poket,81:next
 50 fort=1063to2023step40:poket,81:next
 60 fort=2023to1984step-1:poket,81:next
 70 fort=1984to1024step-40:poket,81:next
100 g=1964:br=40:gr=1:poke1704,32:poke1
    705,32:poke1664,32:poke1665,32
130 ifpeek(g+br)=32thengosub300
131 ifpeek(g+gr)<>32thengosub200:goto13
    0
135 pokeg,32
140 ifg+gr>1024andg+gr<2023theng=g+gr
150 pokeg,3
155 ifg=1704thenend
160 goto130
200 ifbr=40thengr=-40:br=1:return
210 ifbr=1thengr=-1:br=-40:return
220 ifbr=-40thengr=40:br=-1:return
230 ifbr=-1thengr=1:br=40:return
300 ifbr=40thengr=40:br=-1:return
310 ifbr=-1thengr=-1:br=-40:return
320 ifbr=-40thengr=-40:br=1:return
330 ifbr=1thengr=1:br=40:return
```

```
regel 1    156    60    195    160    29
regel 2    23     70    246    200    175
regel 3    128    100    43    210    39
regel 4    143    130    1     220    90
regel 10    112   131    14    230    124
regel 20    39    135    111   300    175
regel 30    214   140    106   310    210
regel 40    50    150    61    320    90
regel 50    63    155    119   330    209
```

C-16 maestro

We zijn de C-16 toch niet helemaal vergeten. Hier een kort leuk demonstratieprogramma van enkele mogelijkheden die de C-16 te bieden heeft. Hoewel het programma minder dan 35 regels telt geeft het grafisch zowel als muzikaal een aardige demo van z'n kunnen. En dat allemaal zonder dat er ook maar een POKE aan te pas komt.

```
5 gosub1000
6 forx=1to9:readn(x):nextx
8 char1,8,1,"****** C-16 maestro *******"
10 vol7
20 do
30 geta$:ifa$=""then30
33 help
35 a=asc(a$):ifa<49ora>57then50
36 n=a-48
40 sound1,n(n),4
45 gshapen$,150,8*(6+(9-n)),4
46 forz=1to50:nextz
47 gshapen$,150,8*(6+(9-n)),4
50 loop until a=32
55 vol0:graphic0:scnclr
60 end
100 data345,383,453,516,571,596
110 data643,685,704
1000 graphic1,1
1010 fory=60to124step16
1020 draw1,100,yto200,y
1030 nexty
1040 a$="fedcbagfe"
1050 forx=1to9:c=13
1060 ifint(x/2)=x/2thenc=14
1070 char1,c,x+6,mid$(a$,x,1),0
1075 char1,c+10,x+6,right$(str$(10-x),1)
1080 nextx
1090 forx=1to8:fory=11to16:draw1,x,y
1095 nexty,x
1100 y=1:x=8:draw1,8,16tox,y
1110 sshapen$,1,1,8,16
1120 gshapen$,1,1,4
1130 return
```

Vic-invaders

Een spel voor de VIC-20 bezitters. Het bijna klassiek geworden 'Space-invaders' in een gloednieuwe vorm. De listing bestaat uit twee delen, die stuk voor stuk moeten worden geladen en gerund. Omdat het spel voor een groot gedeelte is geschreven in begrijpelijk Basic, is het zondermeer aan je eigen wensen aan te passen. Een knap stukje werk van Paul Prange.

```
   1 rem vic-invaders
   2 rem geen extra geheugen nodig.
   3 rem door paul prange,
   4 rem vaals / 04454-3630
   5 rem
   6 rem matrix van          *** invaders
     *
   7 print"[CLR-HOME]":poke36879,8:poke5
     4,28:poke56,28:restore
  10 form=7168to7175:readd:pokem,d:next
  20 data24,24,60,60,126,102,195,195
  30 form=7176to7335:pokem,0:next
  40 form=7336to7343:pokem,24:next
  50 form=7360to7399:readd:pokem,d:next
  55 data24,60,90,255,165,153,165,165
  60 data24,60,235,189,24,36,36,195
  65 data108,146,56,84,124,124,68,102
  70 data2,40,176,60,126,102,195,195
  75 data64,8,42,217,68,148,113,4
  80 poke36869,255:print"[GEEL]"
  82 forb=7680to8185:pokeb,19:next
  84 print"[op]xsxssxsxsssxssxssxx"
  86 print"ysyysysysssysysysysy"
  88 print"zszszzsszszsszzzszsz"
  90 print"xsxssxssxsxssxsxsxsx"
  92 print"ysyssysssysssysysyys"
  94 print"[neer]szzszzzsszz"
  96 print"xsssxsxsxss"
  98 print"yyysyysssys"
 100 print"zssszszsssz"
 102 print"sxxsxsxsxxs"
 120 rem machine-taal +
 125 form=7400to7565:readd:pokem,d:next
 130 data173,31,145,170,41,16,240,96
 135 data138,41,32,240,23,138,41,8,240,1
     8
 140 data169,127,141,34,145,173,32,145,1
     62,255,142,34,145,41,128,240,7,96
 145 data169,1,141,140,29,96
 150 data173,205,31,105,233,144,5,169,1,
     141,141,29
 160 data162,255,189,205,30,168,24,105,2
     33,144,9,152,157,206,30
 170 data169,19,157,205,30
 180 data202,240,3,76,32,29
 190 data162,184,189,21,30,168,24,105,23
     3,144,9
 195 data152,157,22,30,169,19,157,21,30,
     202,240,3,76,58,29,96
 200 data24,173,22,30,105,233,144,1,96
 205 data162,0,189,23,30,168,24,105,233,
     144,9
 210 data152,157,22,30,169,19,157,23,30
 215 data189,242,30,168,24,105,233,144,9
 220 data152,157,241,30,169,19,157,242,3
     0
 225 data232,138,105,36,240,3,76,94,29,9
     6,0,0
 230 form=7566to7586:readd:pokem,d:next
 235 data24,162,255,189,205,30,105,232
 240 data144,4,142,140,29,96
 245 data202,240,3,76,145,29,96
 250 form=7587to7602:readd:pokem,d:next
 255 data135,159,163,175,159,135,159,147
     ,80,46,80,82,65,78,71,69
 300 m=7176
 310 fork=1to20:readd:fort=32768+8*dto32
     775+8*d:pokem,peek(t):m=m+1:nexttt,k
 315 data4,18,21,11,15,16,16,12,1,25,9,1
     9,12,15,1,4,9,14,32,7
 320 ifpeek(37151)and64=0then330
 325 print"[HOME][WIT][13xneer]abcdsefsg
     hij":wait37151,64,64
 330 print"[op]ssssssssssss":print"[op]k
     lsmnopqrt"
 335 a$="[neer]load"+chr$(13)+"run"+chr$
     (13):poke198,10:form=1to10:poke630+
     m,asc(mid$(a$,m,1))
 340 next
 345 form=7988to8185:pokem+30720,0:pokem
     ,32:next
 350 print"[4xop][ZWART]"

*** EINDE LISTING ***
```

```
      LISTING programma : vic-invaders 2

1004 rem vic-invaders2
1005 poke36869,242:q=1:x=8153:w=24
1040 print"[CLR-HOME][WIT][5xspatie]SET
     LEVEL":print"[5xspatie]¯¯¯¯¯¯¯¯¯
     [neer]"
     :print"[2xneer]f1...Beginner"
1045 print"[neer]f3...Amateur":print"
     [neer]
     f5...Profie":print"[neer]f7...Exper
     t":print"[2xneer]Met Shift..Willeke
     urig"
1050 a=peek(203):b=peek(653)
1055 ifa=39thens=10:m=0:max=1:tg=4:goto1
     080
1060 ifa=47thens=6:m=44:max=2:tg=5:goto1
     080
1065 ifa=55thens=4:m=0:max=3:tg=4:goto10
     80
1070 ifa=63thens=3:m=44:max=4:tg=5:goto1
     080
1075 goto1050
1080 ifb=1thenq=0:s=s+3
1085 poke36869,255:print"[CLR-HOME]":for
     t=7680to8185:poket,19:next:poke7565
     ,0:ib=0:pokex,0
1100 forb=7703to7921+msteps
1105 ifq>0thenq=int(rnd(1)*tg)+1:onqgosub
     1135,1140,1145:next:goto1200
1110 pokeb,w
1115 ifh=1thenw=w+1:ifw>26thenw=24
1120 ifh=1thennext:goto1200
1125 if(b-7680)/22=int((b-7680)/22)thenw
     =w+1:ifw>26thenw=24
1130 next:goto1200
1135 pokeb,24:return
1140 pokeb,25:return
1145 pokeb,26:return
1200 sys7400:ifpeek(7565)=1andib<>1thena
     k=1:p=p+300:f=2:goto1600
1207 ifib=1andx=id-1thenvp=2
1209 ifib=1andx=id+1thenvp=1
1210 onvpgosub1220,1230:goto1240
1220 x=x+1:ifx>8163thenx=x-1:return
1225 pokex-1,19:pokex,0:return
1230 x=x-1:ifx<8142thenx=x+1:return
1235 pokex+1,19:pokex,0:return
1240 ifib<>1then1280
1245 t=peek(id):ift<24then1255
1250 goto1265
1255 ifpeek(id+1)>23thenid=id+1:goto1245
1260 id=id-1:goto1245
```

```
1265 id=id+22:ifid>8185thenib=0:y=id-22:
     pokey,28::ak=2:goto1500
1270 pokeid-22,19:pokeid,t
1275 ifx=idthenak=1:y=x:pokey,27:goto150
     0
1277 goto1290
1280 ifpeek(7564)=1thensys7566:id=7885+peek
     (7564):ib=1:goto1245
1290 tv=int(rnd(1)*(tg)):ifd=>maxthen130
     5
1295 iftv=1thend=d+1:d(d)=x-22
1300 ifd=0then1350
1305 forb=1tod
1310 d(b)=d(b)-22:ifd(1)<7680thenpoked(1
     )+22,19:goto1370
1315 ifpeek(d(b))>23theny=d(b):gosub1800
     :pokey+22,19:goto1500
1317 ifpeek(d(b)+22)>23theny=d(b)+22:gosub
     1800:goto1500
1320 poked(b)+22,19:poked(b),21:next:goto
     1350
1340 fort=1tod-1:d(t)=d(t+1):next:d=d-1:
     b=b-1:next
1350 iftv=2thenvp=1
1355 iftv=3thenvp=2
1360 iftv=4ortv=5thenvp=0
1365 poke7565,0:poke7564,0:ifpeek(653)<>
     5then1200
1367 run
1370 ifd=1thend(1)=0:d=0:goto1350
1375 goto1340
1500 z=peek(y):ifak=0thenpokey,28
1520 poke36877,140+tv:fort=0totv:poke368
     78,t:poke36877,peek(36877)+4
1522 iftv<7andtv>0thenpoke30720+y,tv+1
1523 next
1525 poke36877,0:fort=0to7:poke7672+t,peek
     ((7168+z*8)+t):next
1530 pokey,63:poke36876,135+d+tg+tv
1535 fort=7to4step-1:poke36878,t-4:poke3
     6876,peek(36876)+tv+t
1550 poke7672+7-t,0:poke7672+t,0:forr=1to
     18:nextr,t:pokey,19
1560 pokey+30720,1:poke36876,0:r=0:t=0:y
     =0:u=u+1:ifu>9thenip=ip+50:f=0:goto
     1600
1562 ifak=2thenak=0:goto1280
1563 ifak=1thenp=p+50:ib=0:e=e+1:ife>5then
     f=1:goto1600
1565 ifak=0thenip=ip+10:goto1320
1570 ak=0:goto1280
1600 poke36869,240:print"[CLR-HOME]compu
     ter..."ip:print"[neer]speler....."p
     :poke36877,0
1605 poke36878,6:fort=7587to7594:poke368
     74+f,peek(t):forr=1to400:nextr,t:poke
     36874+f,0
1710 ifak=1thenak=0:max=max+1:tg=tg-1:if
     tg<5thentg=4
1715 d=0:e=0:u=0:h=1:goto1085
1800 ifid=ythenib=0
1805 return
```

Vingervlug

Om goed en snel te kunnen leren typen, kun je dit programmaaatje van Hans Kriele gebruiken. Het is zeer eenvoudig in z'n opzet, en zou haast een miniatuurtje kunnen zijn. Je krijgt een aantal letters op het scherm, die je na moet typen. Na afloop wordt het aantal foute aanslagen en de tijd die je nodig had om alles te typen op het scherm gezet.

```
  1 rem vingervlug / commodore-64
  2 rem door hans kriele
  3 rem spijkenisse / 01880-18340
  4 rem
 10 kl=53280:pokekl,0:pokekl+1,0:print"
    [CLR-HOME]"
 20 input"[GEEL]hoeveel letters wil je"
    ;l
 30 print"[CLR-HOME]":ti$="000000"
 40 x=int(rnd(1)*26+1)
 50 d=d+1:ifl=d-1thengoto120
 60 printspc(15)chr$(x+64)
 70 poke198,0:wait198,1
 80 geta$
 90 ifa$=chr$(x+64)thenprint"[CLR-HOME]
    [GEEL][9xneer]"
100 ifa$<>chr$(x+64)thenprint"[CLR-HOME]
    [WIT][9xneer]"
110 goto40
120 print"[neer] je hebt"l"letters met"
    sc"fout in deze[4xspatie]tijd: "ti$
    "
130 print"[2xneer][8xrechts]wil je nog
    eens?"
140 geth$
150 ifh$="j"thenrun
160 ifh$="n"thenprint"[CLR-HOME]":end
170 goto140

*** EINDE LISTING ***
```

regel	1	3	regel	80	6
regel	2	169	regel	90	116
regel	3	52	regel	100	141
regel	4	143	regel	110	237
regel	10	88	regel	120	88
regel	20	148	regel	130	176
regel	30	129	regel	140	13
regel	40	60	regel	150	104
regel	50	187	regel	160	12
regel	60	82	regel	170	30
regel	70	96			

Dubbel-border (versie 2)

Al eerder is in deze rubriek een subroutine geplaatst om de schermrand te voorzien van twee kleuren. Dat was een briljant idee, maar minder goed uitgevoerd, in verband met video-flikkeringen etc. Daarom een tweede versie, die deze problemen geheel ondervangt. Alleen bij het laden en saven krijg je nog psychedelische effecten op het scherm (helaas). Zie voor gebruik de regels 40 - 70.

```
LISTING programma : dubbelborder

  1 rem dubbelborder v2.0 / cbm-64
  2 rem door leon van der torre
  3 rem zevenhuizen / 01802-1624
  4 rem een verbeterde versie uit c-inf
    o
  5 rem jaargang 1 nummer 5
  6 rem
 10 s=0:restore
 20 forx=49152to49271:reada:pokex,a:s=s
    +a:next
 30 ifs<>14589thenprint"[CLR-
    HOME]fout
    in data":end
 40 sys49152,150,1,2
 50 rem 150 = positie
    scheidingslijn
 60 rem   1 = kleur 1
 70 rem   2 = kleur 2
100 data 32 , 253 , 174 ,
    32 , 158
101 data 183 , 142 , 103 ,
    192 , 32
102 data 253 , 174 , 32 , 158 , 183
103 data 142 , 108 , 192 , 32 , 253
104 data 174 , 32 , 158 , 183 , 142
105 data 95 , 192 , 173 , 21 , 3
106 data 201 , 192 , 208 , 1 , 96
107 data 120 , 169 , 127 , 141 , 13
108 data 220 , 169 , 1 , 141 , 26
109 data 208 , 173 , 103 , 192 , 141
110 data 18 , 208 , 169 , 27 , 141
111 data 17 , 208 , 169 , 69 , 141
112 data 20 , 3 , 169 , 192 , 141
113 data 21 , 3 , 88 , 96 , 173
114 data 25 , 208 , 141 , 25 , 208
115 data 41 , 1 , 240 , 31 , 173
116 data 32 , 208 , 41 , 15 , 205
117 data 95 , 192 , 240 , 13 , 169
118 data 0 , 141 , 18 , 208 , 169
119 data 0 , 141 , 32 , 208 , 76
120 data 49 , 234 , 169 , 25 , 141
121 data 18 , 208 , 169 , 1 , 141
122 data 32 , 208 , 104 , 168 , 104
123 data 170 , 104 , 64 , 234 , 234
```

```
regel 1    121
regel 2    61
regel 3    14
regel 4    171
regel 5    4
regel 6    143
regel 10   251
regel 20   12
regel 30   13
regel 40   32
regel 50   141
regel 60   177
regel 70   179
regel 100  209
regel 101  251
regel 102  8
regel 103  254
regel 104  5
regel 105  110
regel 106  156
regel 107  250
regel 108  150
regel 109  46
regel 110  213
regel 111  218
regel 112  154
regel 113  67
regel 114  203
regel 115  94
regel 116  148
regel 117  215
regel 118  156
regel 119  101
regel 120  214
regel 121  157
regel 122  251
regel 123  252
```

Sprites opnieuw

De zoveelste generator van sprites. Maar wel een listing, die kort genoeg is om op een avond in te toetsen. Verder krijg je een duidelijke (grote) matrix op je scherm, waar je de sprite op kunt ontwerpen. Dit doe je door gebruik te maken van de toetsen : /, ;, z, x, ⋆, spatie, 9, CLR HOME , HOME en d (voor data) en s (stoppen).
Na het intypen van 'd', wordt de ontworpen sprite als data in het programma gezet. Denk er dus aan het originele programma voor gebruik te saven.

```
1 rem spritegenerator / commodore-64
2 rem door rob danen
3 rem den haag / 070-252436
4 rem
8 rem cursor
9 rem ======
10 v=53248:pokev+21,4:poke2042,13
11 forx=0to20:poke832+x,0:nextx
12 forx=0to20step3
13 poke853+x,255:poke853+(x+1),0
14 poke853+(x+2),255
15 nextx
16 forx=0to20:poke874+x,0:nextx
17 sy=v+5:sx=v+4:poke874,255:poke875,2
   55:poke876,255
18 rem bouw scherm op
19 rem ==============
20 x=24:y=64
21 t1$="UCCCCCCCCCCCCCCCCCCCCCCCCI"
22 t2$="B[24xspatie]B"
23 t3$="JCCCCCCCCCCCCCCCCCCCCCCCCK"
24 i$="........................"
25 dims$(24):dimsd(70)
26 forc=1to24:s$(c)=i$:nextc
27 print"[CLR-HOME]":printt1$
28 forc=1to21:printt2$:nextc
29 printt3$
30 print"[HOME][13xneer][27xrechts][RVS-aa
   n] sprite[2xspatie][RVS-uit]"
31 print"[27xrechts][RVS-aan]generator
   [RVS-uit]"
32 print"[27xrechts][RVS-aan]door:[4xspati
   e][RVS-uit]"
33 print"[27xrechts][RVS-aan]rob danen
   [RVS-uit]"
34 print"[HOME][neer]":forc=1to21:print
   "[rechts]"s$(c):nextc:goto55
35 rem menu
36 rem ====
37 print"[HOME][neer]"
38 forc=1to21:print"[rechts]"s$(c):next
   c
39 geta$:ifa$=""then39
40 ifa$="/"theny=y+8
41 ifa$=";"theny=y-8
42 ifa$="z"thenx=x-8
43 ifa$="x"thenx=x+8
44 ifa$="*"then61
45 ifa$=" "then75
46 ifa$="9"then87
47 ifa$="[CLR-HOME]"thengoto26
48 ifa$="[HOME]"thenx=24:y=64:goto55
49 ifa$="d"then95
50 ifa$="s"thenstop
51 ifx<=16thenx=x+8:goto39
52 ifx>=216thenx=x-8:goto39
53 ify<=56theny=y+8:goto39
54 ify>=232theny=y-8:goto39
55 pokesx,x:pokesy,y
56 print"[HOME][2xneer][29xrechts]y=[2xspa
   tie]";(int((y-56)/8));"[2xspatie]"
57 print"[29xrechts]x=[2xspatie]";(int
   ((x-23)/8))+1;"[2xspatie]"
58 goto39
59 rem "Q" plaatsen
60 rem ============
61 s=int((y-56)/8)
62 l=int((x-23)/8)+1
63 ifl=1then71
64 ifl=24then72
65 a$="":forc=1tol-1
66 a$=a$+mid$(s$(s),c,1):nextc
67 a$=a$+"Q"
68 forc=l+1to24
69 a$=a$+mid$(s$(s),c,1):nextc
70 s$(s)=a$:goto37
71 a$="Q":forc=2to24:a$=a$+mid$(s$(s),
   c,1):nextc:s$(s)=a$:goto37
72 a$="":forc=1to23:a$=a$+mid$(s$(s),c
   ,1):nextc:s$(s)=a$+"Q":goto37
73 rem "Q" weghalen
74 rem ============
75 s=int((y-56)/8)
76 l=int((x-23)/8)+1
77 ifl=1then84
78 a$="":forc=1tol-1
79 a$=a$+mid$(s$(s),c,1):nextc
80 a$=a$+"."
81 forc=l+1to24
82 a$=a$+mid$(s$(s),c,1):nextc
83 s$(s)=a$:goto37
84 a$="":a$=".":forc=2to24:a$=a$+mid$(
   s$(s),c,1):nextc:s$(s)=a$:goto37
85 rem "[RVS-aan]" reverse
86 rem ===========
87 forc=1to21:a$="":forc1=1to24:print"
   [HOME][26xrechts]";21-c
88 ifmid$(s$(c),c1,1)="."thena$=a$+"Q"
89 ifmid$(s$(c),c1,1)="Q"thena$=a$+"."
90 nextc1:s$(c)=a$:nextc
91 print"[HOME][26xrechts][3xspatie]"
92 goto34
93 rem omrekenen
94 rem =========
95 print"[CLR-HOME][RVS-aan]data sprit
   e[RVS-uit]":pokev+21,0
96 forc=1to21
97 a$="":forc1=1to8
98 ifmid$(s$(c),c1,1)="."thena$=a$+"0"
99 ifmid$(s$(c),c1,1)="Q"thena$=a$+"1"
100 nextc1:t=0:gosub113
101 a$="":forc1=9to16
102 ifmid$(s$(c),c1,1)="."thena$=a$+"0"
103 ifmid$(s$(c),c1,1)="Q"thena$=a$+"1"
104 nextc1:t=5:gosub113
105 a$="":forc1=17to24
106 ifmid$(s$(c),c1,1)="."thena$=a$+"0"
107 ifmid$(s$(c),c1,1)="Q"thena$=a$+"1"
108 nextc1:gosub113:t=10:print
109 nextc:c2=0
110 goto120
111 rem reken sub-routine
112 rem =================
113 a=0:b=0
114 forc1=8to1step-1:b=b+1
115 a=a+val(mid$(a$,b,1))*2↑(c1-1)
116 nextc1:c2=c2+1:sd(c2)=a
117 print"[HOME][neer]";c2:poke895+c2,a
    :return
118 rem toon sprite + data
119 rem ==================
120 poke2043,14:pokev+23,8:pokev+29,8
121 pokev+21,8
122 print"[CLR-HOME][3xneer][2xspatie]U
    CCCCCCCI"
123 print"[2xspatie]B[7xspatie]B"
124 print"[2xspatie]B[7xspatie]B"
125 print"[2xspatie]B[7xspatie]B"
126 print"[2xspatie]B[7xspatie]B"
127 print"[2xspatie]B[7xspatie]B"
128 print"[2xspatie]JCCCCCCCK"
129 pokev+6,50:pokev+7,80
```

```
130 print"[3xneer]druk een toets voor d
    ata"
131 geta$:ifa$=""then131
132 pokev+21,0:print"[CLR-HOME]"
133 print"[CLR-HOME][neer]naar [RVS-aan]
    s[RVS-uit]cherm of [RVS-aan]p[RVS-uit]
    rinter ?"
134 geta$:ifa$=""then134
135 p$="n"
136 ifa$="p"theninput"device nummer (4,"
    ;d:open4,d:p$="j"
137 forc=1to63step4
138 ifp$="j"thenprint#4,sd(c);tab(10);s
    d(c+1);tab(10);sd(c+2);tab(10);sd(c
    +3)
139 ifp$="j"thenprint#4," "
140 ifp$="n"thenprintsd(c);tab(10);sd(c
    +1);tab(20);sd(c+2);tab(30);sd(c+3)
141 nextc
142 print"[2xneer]druk een toets"
143 geta$:ifa$=""then143
144 print"[CLR-HOME]":pokev+21,4:goto27
```

```
regel 1    120     16   167     28   42
regel 2    12      17   21      29   68
regel 3    160     18   45      30   87
regel 4    143     19   229     31   55
regel 8    109     20   31      32   254
regel 9    253     21   133     33   217
regel 10    203    22   36      34   84
regel 11    161    23   126     35   196
regel 12    157    24   179     36   131
regel 13    252    25   195
regel 14    46     26   20
regel 15    218    27   236
```

```
regel 37    1      72    254    108   198
regel 38    236    73    239    109   86
regel 39    57     74    107    110   28
regel 40    2      75    176    111   65
regel 41    15     76    125    112   156
regel 42    44     77    205    113   129
regel 43    41     78    8      114   132
regel 44    30     79    39     115   108
regel 45    249    80    152    116   49
regel 46    53     81    167    117   124
regel 47    17     82    39     118   235
regel 48    236    83    95     119   217
regel 49    63     84    4      120   43
regel 50    112    85    1      121   94
regel 51    201    86    46     122   150
regel 52    250    87    7      123   97
regel 53    208    88    180    124   97
regel 54    251    89    180    125   97
regel 55    200    90    81     126   97
regel 56    167    91    226    127   97
regel 57    69     92    240    128   199
regel 58    245    93    51     129   250
regel 59    252    94    180    130   13
regel 60    107    95    149    131   98
regel 61    176    96    174    132   0
regel 62    125    97    73     133   165
regel 63    201    98    19     134   101
regel 64    255    99    183    135   184
regel 65    8      100    194   136   156
regel 66    39     101    128   137   145
regel 67    59     102    19    138   7
regel 68    167    103    183   139   34
regel 69    39     104    199   140   175
regel 70    95     105    174   141   197
regel 71    18     106    19    142   156
                   107    183   143   101
                                144   48
```

Huisverbruik

In Urk hoeven ze niet meer wakker te liggen van de energie-crisis. Gerrit Koffeman (uit Urk) heeft ervoor gezorgd, dat van een heel jaar het verbruik van gas, water en licht in de computer kan worden opgeslagen. Je kunt op elk gewenst moment een overzicht van je totale verbruik bekijken. Het programma werkt met bestandsopslag op disk, maar door het aanpassen van regels 480/530 en 610, kun je het simpel geschikt maken voor gebruik met cassette. Device 8 wordt 1. Van de schermopmaak kun je in ieder geval wat leren

```
  1 rem huisverbruik / commodore-64
  2 rem door gerrit koffeman
  3 rem uit urk
  4 rem
 10 dimm$(13),q(4,13)
 20 m$(1)="januari":m$(2)="februari":m$
    (3)="maart":m$(4)="april"
 30 m$(5)="mei":m$(6)="juni":m$(7)="jul
    i":m$(8)="augustus"
 40 m$(9)="september":m$(10)="oktober":
    m$(11)="november":m$(12)="december"
 50 v$(1)="gas":v$(2)="licht":v$(3)="wa
    ter"
 60 op$="[40xspatie]"
 70 rem ............menu...............
 80 print"[CLR-HOME][WIT]":a$="** menu
    **":gosub840
 90 printtab(5)"[3xneer][ 1 ][4xspatie]
    invoer"
100 printtab(5)"[neer][ 2 ][4xspatie]to
    talen"
110 printtab(5)"[neer][ 3 ][4xspatie]pr
    inten"
120 printtab(5)"[neer][ 4 ][4xspatie]sa
    ven / laden"
130 gosub880:ifval(a$)<1orval(a$)>4then
    130
140 onval(a$)goto160,340,680,480
150 rem ..........invoer..............
160 a$="gas, licht, water":gosub840:print
    tab(5)"[3xneer][ 1 ][6xspatie]gas"
170 printtab(5)"[neer][ 2 ][6xspatie]li
    cht"
180 printtab(5)"[neer][ 3[shift-SPATIE]
    ][6xspatie]water"
190 gosub880
200 ifval(a$)<1orval(a$)>3then190
210 u=val(a$):fory=1to250:next
220 a$="invoer[2xspatie]"+v$(u):gosub84
    0:input"[2xneer][2xrechts]welke maa
    nd wenst u [nr.]";m
230 ifm<1orm>12then220
240 printtab(29)"[op][RVS-aan]"m$(m)"[RVS-u
    it]":print"[neer]"tab(15)"o.k. [j/n
    ]":gosub880
250 ifa$="n"then220
260 a$="invoer[2xspatie]"+v$(u)+"[2xspatie]"
    +m$(m):gosub840
270 input"[neer]geef begin meterstand";
    bs
280 input"[neer]geef eind meterstand ";
    es
290 print"[22xspatie]------"
300 print"[8xspatie]verbruik[6xspatie]"
    es-bs
310 q(u,m)=(es-bs)
320 gosub880:goto80
330 rem.........totalen..............
340 gosub940
350 a$="totaal verbruik ":gosub840
360 print"[2xspatie]maand[2xspatie]| [3xspat
    ie]gas[4xspatie]| licht[3xspatie]|
    water"
370 print"----------□-----------□---------
    -□---------"
380 fori=1to12
390 printm$(i)+left$(op$,9-len(m$(i)));
400 print"| ";tab(20)"| ";tab(30)"|  ":next
410 fory=0to39:print"-";:next
420 print"[neer]"tab(12)q(1,0);tab(22)q
    (2,0);tab(32)q(3,0)
430 print"[HOME][5xneer]"
440 fory=1to12:gosub910
450 printtab(12)z$;tab(22)x$;tab(32)c$:
    next:gosub880
460 goto80
470 rem...........laden/saven..........
480 a$="laden en saven":gosub840:open15
    ,8,15
490 input"[2xneer][3xrechts]wilt u [RVS-aan
    ]l[RVS-uit]aden of [RVS-aan]s[RVS-uit]
    aven ";a$
500 ifa$="l"then600
510 ifa$<>"s"thenprint"[4xop]":goto490
520 a$="saven":gosub840:input"[2xneer]n
    aam";h$
530 open2,8,2,"@0:"+h$+",s,w"
540 gosub790
550 foru=1to3:form=1to12
560 print#2,q(u,m)
570 gosub790:next:next
580 close2:close15
590 goto80
600 a$="laden":gosub840:input"[2xneer]n
    aam";h$
610 open2,8,2,h$+",s,r"
620 gosub790
630 foru=1to3:form=1to12
640 input#2,q(u,m)
650 gosub790:next:next
660 close2:close15:goto80
670 rem...........printen..............
    .
680 open4,4:a$="totaal verbruik "
690 print#4,"[12xspatie]"a$
700 print#4:print#4,"[3xspatie]maand[6xspat
    ie]gas[6xspatie]licht[6xspatie]wate
    r"
710 forx=0to39:print#4,"¯";:next:print#
    4
720 fory=1to12:m$(y)=m$(y)+left$(op$,9-
    len(m$(y))):gosub910
730 print#4,m$(y)"[3xspatie]"z$"[6xspatie]"
    x$"[6xspatie]"c$:next
740 fory=0to39:print#4,"-";:next:print#
    4
750 gosub940
760 print#4,"[12xspatie]"q(1,0)"[5xspatie]"
    q(2,0)"[7xspatie]"q(3,0):close4:goto
    70
770 rem ...........gosub..............
780 rem *** error lezen op disk ****
790 input#15,en,em$,et,es
800 ifen=0thenreturn
810 print"[4xrechts] error op de disk "
820 printen;em$;et;es
830 close2:close15:end
840 v=(40-len(a$))/2
850 print"[CLR-HOME][RVS-aan]"op$
860 print"[RVS-aan][op]"left$(op$,v);a$
    ;left$(op$,v+1)
870 print"[RVS-aan][op]"op$"[RVS-uit]":
    return
880 print"[HOME][23xneer][11xrechts][RVS-aa
    n]druk op een toets![RVS-uit]"
890 geta$:ifa$=""then890
900 return
910 g$=str$(q(1,y)):g=len(g$):z$=left$(
    op$,5-g)+g$
920 h$=str$(q(2,y)):h=len(h$):x$=left$(
    op$,5-h)+h$
```

```
930 f$=str$(q(3,y)):f=len(f$):c$=left$(
    op$,5-f)+f$:return
940 foru=1to3:q(u,0)=0
950 forx=1to12:q(u,0)=q(u,0)+q(u,x):next
    :next:return
```

```
huisverbruik

regel 1     157
regel 2     215
regel 3     115
regel 4     143
regel 10     62
regel 20    159
regel 30     46
regel 40     63
regel 50     31
regel 60    185
regel 70    158
regel 80     74
regel 90    205
regel 100   240
regel 110   250
regel 120   235
regel 130   156
regel 140   129
regel 150   224
regel 160   247
regel 170    77
regel 180   253
regel 190    45
regel 200    58
regel 210   112
regel 220   135
regel 230     8
regel 240    22
regel 250   111
regel 260     2
regel 270    29
regel 280   219
regel 290   235
regel 300    31
regel 310    75
regel 320    88
regel 330   200
regel 340    42
regel 350   237
regel 360    44
regel 370     6
regel 380   180
regel 390   148
regel 400    16
regel 410     7
regel 420   132
regel 430    69
regel 440    37
regel 450    35
regel 460   241
regel 470   101
regel 480   103
regel 490    53
regel 500   111
regel 510    19
regel 520    36
regel 530   179
regel 540    45
regel 550   130
regel 560   102
regel 570   165
regel 580    18
regel 590   241
regel 600    11
regel 610    22
regel 620    45
regel 630   130
regel 640    82
regel 650   165
regel 660    61
regel 670    91
regel 680   247
regel 690   161
regel 700   117
regel 710   245
regel 720    12
regel 730   252
regel 740   108
regel 750    42
regel 760   140
regel 770   141
regel 780   135
regel 790    20
regel 800    53
regel 810    46
regel 820   196
regel 830   204
regel 840   192
regel 850    69
regel 860   242
regel 870   225
regel 880   183
regel 890   110
regel 900   142
regel 910   234
regel 920   238
regel 930   152
regel 940   255
regel 950   186
```

Computerster

De reclame-jingle van Loekie de Leeuw komt nu uit de C-64. Als je goed luistert denk je bijna dat het de STER is. Door de data in de regels 100 tot 140 te wijzigen, verandert ook de deun. Denk eraan geen getallen te gebruiken die groter zijn dan 255 of kleiner dan nul.

```
  1 rem computerster / commodore-64
  2 rem door erik hendrikx
  3 rem uit westervoort
  4 rem
 10 fors=54272to54296:pokes,0:next:k=54
    272:pokek+5,0:pokek+6,240:pokek+4,1
    7
 30 pokek+24,15:forz=1to33:reada,b,c:poke
    k+1,a:pokek,b:fort=1toc:next:next
 70 rem
100 data57,172,100,0,0,0,57,172,100,0,0
    ,0,57,172,100,0,0,0,57,172,125
110 data 51,97,250,45,198,125,38,126,25
    0,34,75,125,38,126,250,45,198,125
120 data 51,97,250,57,172,250,0,0,250,5
    7,172,100,0,0,0,57,172,100,0,0,0
130 data 57,172,100,0,0,0,57,172,100,0,
    0,0,61,126,110,68,149,250,57,172,12
    5
140 data 45,198,250,38,126,125,45,198,2
    50,51,97,357,45,198,250,0,0,0

* EINDE LISTING ***
```

```
computerster

regel 1     167
regel 2      75
regel 3     245
regel 4     143
regel 10     82
regel 30    164
regel 70    143
regel 100     6
regel 110   195
regel 120    66
regel 130    20
regel 140   246

ready.
```

Datamaker

Een heel handige utility, waarmee een willekeurig blok computergeheugen in Basic dataregels wordt omgezet. De begin- en eindadressen moeten decimaal worden ingegeven. *De grap, waardoor het programma zichzelf* a.h.w. wijzigt, schuilt in regel 5, waarin door de computer zelf (op het scherm) het commando 'RUN6' wordt gezet en uitgevoerd.

```
0 rem datamaker/cbm-64 door l v/d tor
  re
1 input"[CLR-HOME]begin adres";ba:input
  "eind adres";ea:rn=100
2 print"[CLR-HOME][2xneer]9data";ba;"
  ,";ea;",";rn:print"run4[HOME]";
3 poke632,13:poke633,13:poke634,13:poke
  198,3:end
4 readba,ea,rn:print"[CLR-HOME][2xneer]"
  ;rn;"data";:forx=0to4:printpeek(x+b
  a);",";:next
5 print"[DELete]":print"run6[HOME]";:
  poke632,13:poke633,13:poke634,13:poke
  198,3:end
6 readba,ea,rn:ba=ba+5:rn=rn+1:ifba>e
  athenend
7 goto2

 EINDE LISTING ***        regel 0    134
                          regel 1    94
                          regel 2    35
                          regel 3    73
                          regel 4    221
                          regel 5    4
                          regel 6    6
                          regel 7    187
```

Delete

Een simpele routine, die ervoor zorgt dat regels uit een programma kunnen worden verwijderd, zonder met de cursor eindeloos over het scherm te hoeven bewegen. Het werkt langzaam maar doeltreffend. Je moet voor het gebruik wel de zes programma-regels als eerste regels van je eigen programma intoetsen, anders werkt het niet. Niet de meest elegante oplossing, maar wel kort.

```
0 rem delete/cbm-64 door l v/d torre
1 input"[CLR-HOME]begin regel";br:input
  "eind regel";er
2 print"[CLR-HOME][2xneer]9data";br;"
  ,";er:print"run4[HOME]";
3 poke632,13:poke633,13:poke634,13:poke
  198,3:end
4 readbr,er:print"[CLR-HOME][2xneer]"
  ;br
5 print:print"run6[HOME]";:poke632,13
  :poke633,13:poke634,13:poke198,3:end
6 readbr,er:br=br+1:ifbr>erthenend
7 goto2

 EINDE LISTING ***        regel 0    175
                          regel 1    99
                          regel 2    191
                          regel 3    73
                          regel 4    121
                          regel 5    172
                          regel 6    149
                          regel 7    187
```

Smooth scroll

Het echte smooth scrollen van het scherm was tot nog toe uitsluitend voorbehouden aan de machinetaal-freaken. Maar daar brengt dit programma verandering in. Hiermee *kun je vanuit Basic een deel van het scherm heel mooi* laten scrollen.

Een demo van de werking is ingebouwd. Voordat je er zelf mee aan het werk gaat, verdient het aanbeveling de regels 60 en 70 goed te bestuderen. Nadat dit programma is gerund, blijft het aanwezig, tot je de computer uitzet.

```
  1 rem smooth scroll / commodore-64
  2 rem door leon van der torre
  3 rem zevenhuizen / 01802-1624
  4 rem
 10 s=0
 20 forx=49152to49280:reada:pokex,a:s=s
    +a:next
 30 ifs<>18585thenprint"[CLR-HOME]fout
    in data":end
 40 rem demo
 50 rem sys49152 = scroll
 60 print"[CLR-HOME]";:forx=1to20:print
    "smooth scroll commodore info !":next
 70 forx=1to20:sys49152:next:end
100 data 165 , 172 , 72 , 165 , 173
101 data 72 , 165 , 174 , 72 , 165
102 data 175 , 72 , 162 , 255 , 198
103 data 214 , 198 , 201 , 206 , 165
104 data 2 , 232 , 32 , 44 , 192
105 data 224 , 24 , 176 , 12 , 189
106 data 241 , 236 , 133 , 172 , 181
107 data 218 , 32 , 200 , 233 , 48
108 data 236 , 76 , 19 , 233 , 224
109 data 0 , 208 , 42 , 120 , 169
110 data 6 , 141 , 150 , 192 , 32
111 data 109 , 192 , 173 , 17 , 208
112 data 41 , 240 , 24 , 109 , 150
113 data 192 , 141 , 17 , 208 , 206
114 data 150 , 192 , 16 , 236 , 160
115 data 40 , 204 , 18 , 208 , 208
116 data 251 , 169 , 128 , 44 , 17
117 data 208 , 208 , 242 , 240 , 15
118 data 224 , 13 , 208 , 11 , 173
119 data 17 , 208 , 41 , 240 , 24
120 data 105 , 7 , 141 , 17 , 208
121 data 32 , 240 , 233 , 96 , 160
122 data 2 , 169 , 248 , 205 , 18
123 data 208 , 208 , 251 , 169 , 249
124 data 205 , 18 , 208 , 208 , 251
125 data 136 , 208 , 239 , 96 , 234

     regel 1    163        regel 108   213
     regel 2    61         regel 109   150
     regel 3    14         regel 110   150
     regel 4    143        regel 111   6
     regel 10    53        regel 112   196
     regel 20    12        regel 113   255
     regel 30    13        regel 114   254
     regel 40    180       regel 115   202
     regel 50    159       regel 116   214
     regel 60    21        regel 117   251
     regel 70    21        regel 118   198
     regel 100    9        regel 119   150
     regel 101    217      regel 120   152
     regel 102    16       regel 121   204
     regel 103    51       regel 122   163
     regel 104    101      regel 123   62
     regel 105    212      regel 124   255
     regel 106    48       regel 125   13
     regel 107    201
```

Hollands foutje.

Een stukje Basic, dat kan worden gebruikt als een aardig hulpmiddel bij het programmeren. De standaardmeldingen als 'Syntax error', 'too many files open' en 'division by zero' etc. worden door dit programma nl. vervangen door nederlandse boodschappen. Leuk om te zien, aardig om te hebben en zeker voor kleinere kinderen de moeite waard om te gebruiken. Het programma runnen, daarna 'NEW' en alles kan weer met de 64 (in Basic) worden gedaan.

```
  1 c0=49152:rem$c000
  2 c5=49232:rem$050
  3 c9=49296:rem$c090
  4 rem  768:139->0
  5 rem  769:227->192
  6 rem*
  7 rem* dit is een programma van mark
    westbroek
  8 rem* het maakt het mogelijk de stan
    daard foutmeldingen te veranderen,
  9 rem* en list in programmastand de p
    rogrammaregel te veranderen met de
    fout.
 10 rem*voor de commodore-64
 11 rem
 12 rem
 15 fort=c0toc0+76:reada:poket,a:next
 20 reada$:t=1:iflen(a$)=0then22
 21 fort=1tolen(a$):pokec9+t-1,asc(mid$
    (a$,t,1)):next
 22 pokec9+t-1,0
 30 a=c9+t:b=c5:forf=1to30:c=(a-32768)and
    255:pokeb,c:pokeb+1,(a-c)/256:b=b+2
 35 reada$:fort=1tolen(a$)-1:pokea,asc(
    mid$(a$,t,1)):a=a+1:next
 40 pokea,asc(right$(a$,1))+128:a=a+1:next
 50 poke768,0:poke769,192:rem activeren
     nieuwe foutmeldingsroutine
 90 rem****************************
 91 rem**100-120=machinecode prg.**
 92 rem**150     ="[2xspatie]error"
      **
 93 rem**201-229=foutmeldingen    **
 94 rem****************************
100 data138,16,3,76,116,164,10,170,189,
    78,192,133,34,189,79,192,133,35,32,
    204
110 data255,169,,133,19,32,215,170,32,6
    9,171,160,,177,34,72,41,127,32,71,1
    71
120 data200,104,16,244,32,122,166,169,1
    44,160,192,32,30,171,164,58,200,240
    ,199
130 data165,57,133,20,165,58,133,21,32,
    194,189,32,19,166,76,201,166
150 data
201 datate veel files,file al geopend,f
    ile niet geopend,file niet gevonden
205 dataapparaat niet aanwezig,geen inv
    oer file,geen uitvoer file
208 datageen filenaam,fout apparaatnumm
    er,next zonder for,zinsbouw fout
212 datareturn zonder gosub
213 datageen data meer,foute waarde,ove
    rflow,geheugen vol
217 dataniet bestaande regel,fout array
    -elementnummer,array al gedim'd
220 datadeling door nul,mag niet in dir
    ecte mode
222 datafout type variabele,string te l
    ang,fout in file data,formule te mo
    eilijk
226 datakan niet doorgaan,niet gedefine
    erde funktie,verificatie fout
229 dataleesfout,stop fout

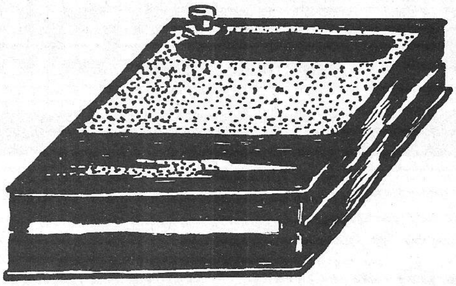
*** EINDE LISTING ***

    hollands foutje          regel 90    39
                             regel 91    236
                             regel 92    216
    regel 1    234           regel 93    162
    regel 2    176           regel 94    39
    regel 3    5             regel 100    233
    regel 4    166           regel 110    157
    regel 5    17            regel 120    250
    regel 6    185           regel 130    245
    regel 7    122           regel 150    131
    regel 8    214           regel 201    230
    regel 9    59            regel 205    42
    regel 10    196          regel 208    118
    regel 11    143          regel 212    181
    regel 12    143          regel 213    77
    regel 15    120          regel 217    166
    regel 20    136          regel 220    70
    regel 21    88           regel 222    186
    regel 22    73           regel 226    54
    regel 30    113          regel 229    154
    regel 35    142
    regel 40    65
    regel 50    197          ready.
```

Wie zelf programma's schrijft, kan die opsturen naar de redactie. We hebben op dit moment een prijsvraag lopen, en om inzenders van eerdere programma's een eerlijke kans te geven, bekijken we ook hun programma's bij de bepaling van de winnaar. Stuur de programma's (of uw commentaar erop naar de redaktie (t.a.v. Rob vd Heuvel of Jan Bodzinga). Vermeld in een briefje wat je met het programma kunt doen, en voor welke computer het geschikt is. We moeten helaas nog steeds een groot deel van de ons aangeboden listings terugsturen, omdat er toch nog (kleine) foutjes in worden aangetroffen, en daar zitten we niet om te springen uiteraard. De gepubliceerde listings zijn door ons meermalen getest en bekeken op bruikbaarheid. Als de listing deze proef doorstaat, besluiten we het programma te plaatsen. Dit wil natuurlijk niet zeggen, dat het 100% zeker is, dat er werkelijk geen fouten in zitten. Maar we doen ons best.
Het heel lange machinetaal-programma 'KOKOS' moet je zien als een uitzondering, vanwege de kwaliteit van deze listing. We gaan de listings in dit nummer overigens weer op band of disk zetten (tezamen met de listings uit het vorige nummer wordt dat dan Infolist no V), zodat iedereen ten volle van dit spel kan genieten. Onze 'INFOLIST-service' kan natuurlijk ook listings uit oudere nummers blijven leveren.

Jan Bodzinga.

Wie veel programmeert, krijgt behoefte aan wat hulpfunkties. Een handige oplossing daarvoor zijn de insteekmodules of Toolkits. Ze zijn er van verschillende huize, met waarschijnlijk Kolff's TAPE QUEEN en DISK SPEEDLOADER als meest bekende, maar ook FLIP's toolkit is bij velen een onmisbaar stukje gereedschap, terwijl in modules zoals RADARBASIC 50KB ook al een aantal toolkitfunkties zitten.



Hulp-modules voor de C-64

We gebruiken zelf ook verschillende van die toolkits en eerlijk gezegd doen ze het allemaal heel goed en worden echte verbeteringen ook snel overgenomen. Wel moet u er om denken, dat de ontwikkeling zeker niet stilstaat en men dus goed op moet passen welke versie men koopt.

Share Operating Service, kortweg SOS, brengt een hele serie nuttige hulpmiddelen op de markt onder de merknaam WEKA, al hebben we gehoord dat men nu naar een andere naam zoekt.

Deze toolkits zijn allemaal bestemd voor de Commodore-64, en worden aangesloten op de expansie-poort.

De verzorging aan de buitenkant is deugdelijk en fraai, het ding past goed en laat zich zonodig gemakkelijk verwijderen. De werking van de door ons bekeken tools is redelijk tot goed. Alle ingebouwde funkties doen waar ze voor staan, en ook de snelheid waarmee dit gebeurt is te aanvaarden.

In feite is er uitgegaan van een hele serie hulpcommando's, die op verschillende manieren zijn gekombineerd, zodat de gebruiker de voor hem juiste selectie zonder moeite kan bepalen.

Hoofdfunkties onderscheiden zich in het gebruik van disk of tape en daarnaast het al of niet gebruiken van machinetaal of eventueel een Centronics-printer.

Standaard programmeerfunkties zijn o.a. AUTO,DEL,DEC,DUMP & FIND, HELP,HEX,KEY OLD en PLIST. Daarnaast een snellader/saver voor disk (6 maal) of cassette (10 maal) en voor de disk alle funkties die horen bij Basic 4.0, dus b.v. Dload en Dsave. De machinetaal monitor heeft meer dan 30 commando's en dat zijn er 16 meer dan gebruikelijk, al zitten er heel eenvoudige tussen.

Verder zijn alle modules uitgerust met een routine die het afregelen van de recorderkop simpel maakt, en dat is vrij bijzonder.

Voor alle modules geldt verder, dat er geen Basic RAM wordt gebruikt. Het pakket voorziet ook in een Nederlandse handleiding (beknopt maar duidelijk) en de apparaatjes zijn getest.

De prijzen lopen uiteen van *f* 79,- voor de simpele turbotool, tot bijna het dubbele (*f* 132,50) maar dan is deze turbootool gekombineerd met de MT-monitor, die op z'n beurt apart *f* 82,50 kost.

Al met al een hele serie nieuwe artikelen, waarmee de mogelijkheden voor de liefhebbers van 'toolkits' weer eens zijn uitgebreid.

J.B.

Een aantal maanden geleden (jaargang 1, no. 4) gaven wij een aantal suggesties om de presentatie van tekst op het scherm aantrekkelijker te maken. Een programma kan nog zo slim, knap of ronduit geniaal zijn, de waarde voor de gebruiker hangt altijd ook af van de 'opmaak'. Een bekend culinair uitgangspunt.
In dit artikel geven we enkele gekompliceerde programma's en ideeën om die "finishing touch" aan te brengen.

SCHERMWERK

Verlevendig de tekst op het beeldscherm

Bijdragen

We zullen het nu niet hebben over de betrekkelijk eenvoudige tips en programma's zoals centreren, brede letters, kleur, reverse, wachtlussen en beeldeffekten (in de rubriek Miniatuurtjes staan daar geregeld leuke voorbeelden van).In plaats daarvan zullen we een programma bekijken om grote letters te maken en een om de letters te laten neerdalen.We hebben daarvoor de gewaardeerde hulp gekregen van Ben van Volen (opnieuw) en Antoine van de Berg (eveneens een bekende voor Commodore-Info-lezers als winnaar van onze eerste prijsvraag in 1984).

Grote letters

Ben van Volen heeft zijn programma voor brede letters aangevuld met een versie voor grote letters. Het verdubbelt de letters ook in hoogte.

Error, error

In een vorig artikel (jaargang 1, no. 4) over schermverfraaiing schreven we over de POS-funktie. De vermelde werking hiervan blijkt echter 'wishful thinking' te zijn geweest. De funktie leest de positie van de cursor, maar kan hem er niet op de beschreven wijze heen sturen. Jammer, maar waar. Ons past een deemoedig ON ERROR GO AND BE ASHAMED.
De onvolkomenheid was overigens niet zozeer een fout, als wel het weglaten van een verklarend zinnetje bij Ben van Volen's programma voor brede letters op het scherm. Daarom nu de correcte gebruiksaanwijzing: de brede letters moeten opgeroepen worden met twee toetsindrukken per karakter. Eerst gewoon "+" en dan 'geshift'. Oplettende lezers hebben ons hierop attent gemaakt.

```
      LISTING programma : 1520 lister   pag. 1
100 rem ******************************
110 rem *                            *
120 rem *         1520 lister        *
130 rem *                            *
140 rem *     door ben van volen     *
150 rem *                            *
160 rem ******************************
170 rem *                            *
180 rem * dit programma drukt op een *
190 rem * 1520 plotter-printer z.g   *
200 rem * 'condensed characters' af, *
210 rem * d.w.z: 'vette letters'.    *
220 rem *                            *
230 rem * de basic-listing moet eerst*
240 rem * weggeschreven worden op de *
250 rem * volgende wijze :           *
260 rem *                            *
270 rem * open2,8,2,"@:listing,u,w"  *
280 rem * (return indrukken)         *
290 rem * cmd2:list                  *
300 rem * (return indrukken)         *
310 rem * print#2:close 2            *
320 rem * (return indrukken)         *
330 rem *                           .*
340 rem * vervolgens kan men 1520    *
350 rem * lister inladen en runnen   *
360 rem *                            *
370 rem ******************************
380 print:print"stop de schijf met het
    programma"
390 print"'listing' erop in de drive,"
400 print"doe de printer aan,
410 print"en druk een toets"
420 print:print
430 geta$:ifa$=""then430
440 open2,8,2,"listing,u,r":open15,8,15
450 open4,4:open1,4,1
460 get#2,ln$
470 ifstthen540
480 print#4,ln$;
490 ifln$=chr$(13)orln$=chr$(32)then460
500 print#1,"r 1,0"
510 print#4,ln$;
520 print#1,"r 11,00"
530 goto 460
540 input#15,a,b$,c,d:printa;b$;c;d
550 close1:close2:close4:close15

*** EINDE LISTING ***
```

Neerdalende letters

Antoine van de Berg maakte op ons verzoek een programma waarbij de letters langzaam van boven uit het beeld komen zakken en op de gewenste plaats stoppen. Mits met mate toegepast, zal ook dit 'grapje' het programma levendig(er) kunnen maken.

Zichzelf schrijvende letters

Graag hadden wij u ook een programma voorgetoverd van 'zichzelf schrijvende letters'. We komen helaas niet verder dan het voorleggen van onze ideeën hierover. De letters zouden (eventueel in kleuren) geleidelijk op het scherm moeten verschijnen, als werden ze geschreven. Er zou ook naar toepassingen van dit effekt gezocht moeten worden in de sfeer van het (ondersteunen van) schrijfonderwijs.

Vandaar ons verlanglijstje:

1. de letters moeten zoveel mogelijk overeenkomen met de gangbare schrijfletters.De letters en cijfers moeten vrij oproepbaar zijn, dus niet alleen vaste teksten;
2. toepassing van verschillende kleuren;
3. het formaat van de letters moet redelijk groot zijn (veel tekst zul je dus niet kwijt kunnen);
4. er moet tenminste één versie bij zijn waarvan de schrijfbewegingen overeenkomen met die van de hand bij het schrijven.

Al met al niet echt eenvoudig en geen programma dat in een handomdraai klaar is, lijkt ons. We zijn benieuwd.

De lichtkrant

Hoewel de lichtkrant al een bekend verschijnsel is, lijkt het ons toch leuk ook hier de volgende keer een voorbeeld van op te nemen. Het programma moet dan natuurlijke wel wat meer in zijn mars hebben dan de tot nu toe gepubliceerde. Wensen:

1. tekst moet vrij in te voeren zijn;
2. minimale tekstlengte 100 karakters;
3. kleur moet vrij programmeerbaar zijn;
4. blijvend vaste tekst boven en/of beneden de lichtkrant moet eveneens vrij ingevoerd kunnen worden;
5. lichtkrant en/of vaste tekst(en) zouden eventueel in grote(re) karakters moeten kunnen verschijnen;
6. klok in beeld hoeft niet, maar zou wel leuk zijn;
7. geluid erbij heeft slechts beperkte betekenis, misschien een leuke aandacht-trekker telkens bij het langskomen van het begin van de tekst;
8. moeilijker, maar misschien wel aardig is een diagonaal over het beeldscherm (linksonder naar rechtsboven) lopende tekst. Dit vergt wellicht te veel van de programmeur, de geheugenruimte of de mogelijkheden van de C-64.

We zijn benieuwd of lezers de uitdaging aannemen. Ook deeloplossingen kunnen een waardevolle bijdrage vormen. Alle bruikbare en gepubliceerde inzendingen worden gehonoreerd.

Kees van der Vlies

```
          LISTING programma : grote letters  pag. 1

    100 v=53248:w=14336:u=512+w:xx=u+504:yy
        =xx+512:goto200
    110 fort=7to1step-1:d=d/2:b$(t)="00"
    120 ifd<>int(d)thenb$(t)="11":d=int(d)
    130 next
    140 b$(0)="11":ifd=0thenb$(0)="00"
    150 forx=0to3:d$(1)=d$(1)+b$(x):next
    160 forx=4to7:d$(2)=d$(2)+b$(x):next
    170 forq=1to2:forx=7to0step-1:c=c+1
    180 d(q)=d(q)+val(mid$(d$(q),c,1))*2↑x
    190 next x:c=0:d$(q)="":nextq:return
    200 poke56334,peek(56334)and254
    210 poke1,peek(1)and251
    220 foraa=0to63:forbb=0to7step2
    230 d=peek(v+aa*8+bb/2):gosub110
    240 pokew+aa*8+bb,d(1):pokew+aa*8+bb+1,d(1)
    250 pokeu+aa*8+bb,d(2):pokeu+aa*8+bb+1,d(2)
    260 d(1)=0:d(2)=0:nextbb
    270 forbb=8to15step2:d=peek(v+aa*8+bb/2
        ):gosub110
    280 pokexx+aa*8+bb,d(1):pokexx+aa*8+bb+
        1,d(1)
    290 pokeyy+aa*8+bb,d(2):pokeyy+aa*8+bb+
        1,d(2)
    300 d(1)=0:d(2)=0:nextbb:nextaa
    310 poke1,peek(1)or4
    320 poke56334,peek(56334)or1:pokev+24,31
    330 forbb=8to15:pokexx+32*8+bb,255:poke
        xx+32*8+bb+1,255:next
    340 forbb=8to15:pokeyy+32*8+bb,255:poke
        yy+32*8+bb+1,255:next
    350 print"[WIT]bBeEnN vVaAnN vVoOlLeEnN.¬"
    360 print"[RVS-aan]bBeEnN[rechts]vVaAnN
        [rechts]vVoOlLeEnN.¬[l.blauw]"

*** EINDE LISTING ***
```

```
          LISTING programma : brede letters  pag. 1

      1 v=53248:w=14336:u=512+w:goto11
      2 fort=7to1step-1:d=d/2:b$(t)="00"
      3 ifd<>int(d)thenb$(t)="11":d=int(d)
      4 next
      5 b$(0)="11":ifd=0thenb$(0)="00"
      6 forx=0to3:d$(1)=d$(1)+b$(x):next
      7 forx=4to7:d$(2)=d$(2)+b$(x):next
      8 forq=1to2:forx=7to0step-1:c=c+1
      9 d(x)=d(x)+val(mid$(d$(x),c,1))*2↑x
     10 next x:c=0:nextq:return
     11 poke56334,peek(56334)and254
     12 poke1,peek(1)and251
     13 fora=0to512:d=peek(a+v):gosub2
     14 pokea+w,d(1):pokea+u,d(2):next
     15 poke1,peek(1)or4
     16 poke56334,peek(56334)or1:pokev+24,31
     17 print"[WIT]bBeEnN vVaAnN vVoOlLeEnN
        .¬[neer][l.blauw]"

*** EINDE LISTING ***
```

Schermwerk (vervolg)

Antoine v/d Berg uit Eindhoven maakte bij het voorgaande artikel een paar programma's voor schitterende tekst, en wel op machinetaal niveau. Dat betekent wel wat Data-regels, maar het is de moeite waard! Ze bestaan steeds uit een Basic POKER en ze zijn eventueel tegelijk te gebruiken, de adressen overlappen elkaar niet. Saven van de routines in machinetaal kan met een Monitor programma.

Midden op het scherm wordt zo een tekst neergezet. Er is voor de aansturing van de machinetaalroutine wel een apart programma nodig, waarbij xps de X-postitie aangeeft, met display komt de SYS, waarmee je het programma aanroept, kleur is de geheugenplaats waar je de gewenste kleur kunt aangeven. Een string (tekstje) in regel 10 moet altijd met een 'nul' karakter worden afgesloten. Deze a$ mag niet langer dan 40 zijn. startadres is $C000, eindadres $C14F

Stuurprogramma 1

```
1 REM /* AANSTURING PROGRAM 1 */
2 REM /*    SCHERM-TEKST      */
3 REM
5 XPS=49176
6 DISPLAY=49152
7 KLEUR=49194
8 :
10 A$="SCHERMTEKST DIE LOKT !"+CHR$(0)
14 :
15 REM /* PUT TEKST (A$) IN MEMORY */
20 FOR X=0 TO LEN(A$)-1
30 POKE 49488+X,ASC(MID$(A$,X+1,1))
35 NEXT
36 :
37 GOSUB 100      :REM /* CENTREER TEKST */
38 POKE KLEUR,15:REM /*DEFINE COLOR */
40 SYS DISPLAY  :REM /* SHOW TEKST   */
45 END
99 :
100 LNGHT=LEN(A$)
110 POKE XPS,((40-LNGHT)/2)
120 RETURN
READY.
```

SCHERMTEKST 1

```
8 RESTORE
9 FOR X=0TO336
10 READ A
11 POKE49152+X,A
12 NEXTX
13 END :REM /* END OF POKE PROGRAM */
14 REM
15 REM /* DATA FOR MACHINE-CODE PROGRAM */
21 DATA32,209,192,169,147,32,210,255,169,15,141,24,212,169,240,141,6,212
22 DATA169,129,141,4,212,169,8,133,253,169,0,133,255,133,249,169,24,133
23 DATA254,169,96,133,250,169,15,32,0,193,166,255,32,245,192,169,102,164
24 DATA253,145,251,232,32,245,192,169,102,145,251,166,254,32,245,192,169
25 DATA102,164,253,145,251,202,32,245,192,163,102,145,251,162,15,160,192
26 DATA136,208,253,202,208,248,165,250,141,1,212,56,233,4,133,250,166,255
27 DATA32,245,192,169,32,164,253,145,251,232,32,245,192,169,32,145,251,134
28 DATA255,166,254,32,245,192,169,32,145,251,202,32,245,192,169,32,145,251
29 DATA134,254,224,12,208,154,164,249,185,80,193,240,45,32,18,193,144,2
30 DATA169,32,166,253,157,224,5,169,96,133,250,230,249,162,0,134,255,169
31 DATA24,133,254,163,128,141,4,212,230,253,202,10,208,252,169,129,141,4
32 DATA212,76,46,192,163,0,141,4,212,141,24,212,96,162,0,138,133,251,169
33 DATA4,133,252,165,252,157,96,3,165,251,157,64,3,24,105,40,133,251,165
34 DATA252,105,0,133,252,232,224,27,208,230,96,189,64,3,133,251,189,96,3
35 DATA133,252,96,162,0,157,0,216,157,0,217,157,0,218,157,0,219,202,208
36 DATA241,96,133,2,48,21,201,32,48,50,201,64,16,3,76,72,193,201,96,16,39
37 DATA56,233,64,76,72,193,165,2,41,127,201,32,48,25,201,64,16,8,165,2,56
38 DATA233,64,76,72,193,201,96,16,9,165,2,56,233,128,24,76,77,193,56,96
39 DATA255,127,72
READY.
```

Dit opgloeiprogramma bestaat eigenlijk uit twee routines, namelijk voor kleur en monochroom. Met kleurflow wordt de juiste SYS opgeroepen en het aantal malen oplichten is met nr1 (regel 20) te doen. In zwartwit dan SYS blakwhite en nr2 voor het aantal malen oplichten. Startadres $C200, eindadres $C2FF

Stuurprogramma 2

```
1 REM /* AANSTURING PROGRAM 2 */
2 REM /*    SCHERM-TEKST      */
3 REM
5 KLEURFLOW=49664
6 BLCKWHITE=49740
7 NR1=49669
8 NR2=49745
9 :
12 PRINT"█████████████ SOFT CONTROL PRESENTS SCHERM-TEKST "
14 POKE 53280,0:POKE53281,0
15 REM /* ZWART SCHERM GEF DE BESTE EFFECTEN !! */
20 POKE NR1,4  :REM /* AANTAL MALEN DAT DE TEKST OPLICHT */
30 SYS KLEURFLOW  :REM /* DO COLORFLOW */
40 END
50 REM /* VOOR ZWART-WIT FLOW 'COLORFLOW' DOOR 'BLCKWHITE' VERVANGEN */
60 REM /* EN 'NR1' DOOR 'NR2' VERVANGEN */
READY.
```

SCHERMTEKST 2

```
5 RESTORE
6 FOR X= 0 TO 120
7 READ A
8 POKE49664+X,A
9 NEXT
10 END:REM /* END OF POKE PROGRAM */
11 DATA169,8,133,252,169,10,133,251,198,252,48,21,166,252,189,42,194,32
12 DATA58,194,162,255,160,86,136,208,253,202,208,248,76,8,194,169,8,133
13 DATA252,198,251,208,223,96,3,7,4,2,6,4,7,1,0,11,12,15,1,15,12,11,162
14 DATA0,157,0,216,157,0,217,157,0,218,157,0,219,202,208,241,96,169,8,133
15 DATA252,169,15,133,251,198,252,48,21,166,252,189,50,194,32,58,194,162
16 DATA255,160,86,202,208,253,136,208,250,76,84,194,169,8,133,252,198,251
17 DATA208,223,96,0,0,0
READY.
```

Deze routine maakt een eigen karakterset en vraagt daarvoor dan ook 4K van het Basic geheugen. Hiermee worden reverse karakter van boven naar beneden gescrolld. Voor het Basic programma geladen wordt, eerst ruimte maken door de volgende POKE's, waarmee de Basic Ram allocatie verandert: POKE 44,24 : POKE 46,24 : POKE 6144,0 : CLR : NEW
Startadres is $C300, eindadres $C420. Geen paniek als de cursor verdwijnt, dit hoort zo.

Stuurprogramma 3

```
1 REM /* AANSTURING PROGRAM 3 */
2 REM /*    SCHERM-TEKST      */
3 REM
6 SETUP=49920
7 SCROLL=49995
8 VRTRGNG=50119
10 REM
20 REM /* ZORG ERVOOR DAT HET BASIC GEHEUGEN OPGESCHOVEN
   IS !! */
30 :
40 SYS SETUP :REM /* SCHUIF CHAR SET OP */
42 POKE 53280,0:POKE53281,0
44 POKE VRTRGNG,130
50 PRINT"█████╭─────────────────────╮"
60 PRINT"████|█SOFT CONTROL WAS HERE█|"
70 PRINT"████╰─────────────────────╯
80 SYS SCROLL
READY.
```

SCHERMTEKST 3

```
5 RESTORE
6 FOR X=0TO240
7 READ A
8 POKE49920+X,A
9 NEXT
10 END:REM /*END OF POKE PROGRAM */
11 DATA120,165,1,41,251,133,1,162,0,189,0,208,157,0,8,189,0,209,157,0,9
12 DATA189,0,210,157,0,10,189,0,211,157,0,11,189,0,213,157,0,13,189,0,214
13 DATA157,0,14,189,0,215,157,0,15,202,208,211,32,209,195,165,1,9,4,133
14 DATA1,88,169,128,141,145,2,169,18,141,24,208,96,162,0,134,251,134,250
15 DATA32,219,195,169,7,133,252,165,250,133,251,164,252,177,253,164,251
16 DATA145,111,198,252,198,251,16,242,165,111,24,105,8,133,111,165,253,24
17 DATA105,8,133,253,201,216,208,216,32,198,195,230,250,165,250,201,8,208
18 DATA202,169,7,133,251,32,198,195,198,251,208,249,32,219,195,169,6,133
19 DATA251,169,7,133,252,164,251,177,111,200,145,111,136,136,16,247,165
20 DATA252,201,7,208,5,200,169,0,145,111,165,111,24,105,8,133,111,201,216
21 DATA208,223,32,198,195,198,252,16,216,96,162,151,160,255,136,208,253
22 DATA202,208,250,96,169,0,170,157,8,12,202,208,250,96,169,8,133,254,169
23 DATA0,133,253,169,12,133,112,169,0,133,111,96,0,0,32,0,0
READY.
```



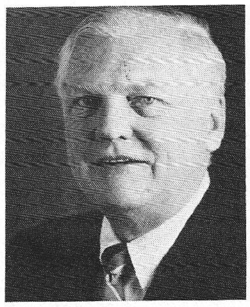
De grote zomertentoonstelling voor consumenteneleketronica, de CES in Chicago, is altijd een ideale plaats om de ontwikkelingen in de VS in ogenschouw te nemen. In de zomer worden daar de spullen aan de handel getoond, die in de decemberperiode in de winkels komen.

De laatste jaren was er altijd een belangrijk deel van die Consumer Electronics Show toebedeeld aan de computers, maar dat bleek dit jaar wat minder geworden. Er is duidelijk een malaise in de hobbycomputermarkt in de VS aan de gang en dat weerspiegeld zich dan in het aantal exposanten. Eigenlijk is het allemaal Commodore, wat de klok in Amerika slaat, een beetje Atari en machines als de IBM PC jr., de Timex (Sinclairs) en zelfs de Apple II blijken wat betreft de hobbycomputers niet meer interessant genoeg, als we kijken naar het softwareaanbod. Ook MSX is in Amerika nog niet van de grond gekomen.

Op zo'n beurs hoef je natuurlijk niet te zoeken naar Nederlandse software, al was AACKOSOFT wel met een fraaie stand aanwezig. Dat bedrijf, erg actief met export bezig, trok met een paar interessante produkten wel de aandacht, en we hopen dat de multimachine en multiplayer actiespelen als "JET-FIGHTER", die daar voor MSX getoond werden, ook in een Commodore versie uitkomen. Aackosoft was al eerder op de CES en we hopen, dat men met de produkten, die kwalitatief de laatste tijd wel vooruitgegaan zijn, ook in Amerika veel succes zal hebben.
De meeste software op de CES is voor de Amerikaanse markt, en zal uiteindelijk ook Europa wel bereiken, maar dat duurt nog enige maanden. Toch is het leuk om nu al iets te vermelden over de hits van dit najaar. Nou ja, hits. als je de leveranciers mag geloven zijn alle programma's absolute toppers en dat kan natuurlijk niet.

Commodore

Op de stand van Commodore op de CES waren natuurlijk vooral de 128's zeer prominent aanwezig. Er was veel software voor, er werd CP/M gedemonstreerd, de 128 werd echt als de opvolger van de 64 naar voren gebracht. Er was ook de 128D, met ingebouwde diskdrive van het 1571 type en die machine is eigenlijk het meest interessant voor onze markt en komt ook in Europa beschikbaar.
Op de CES werden ook de PC 10 en PC 20 en de 900 serie UNIX (Coherent) supermicro's getoond.
Op periferiegebied was er de 1572 dubbele diskdrive met een totale capaciteit van 820 KB, met 340 KB geformatteerd per schijf in de 128 mode. Ook op printergebied was er een nieuwtje met de MPS 1000 printer, een dot matrix eenheid met max. 100 tekens/sec, maar ook een NLQ (Near Letter Quality) mode van 16 t/s en graphics mode met 50 tot 240 dots/inch.
Op een persconferentie onderstreepte Marshall Smith, de president van CBM, dat nog eens. Hij verwachtte nog dit jaar 1 miljoen C-128's te verkopen. Dat werd mede gebaseerd op onderzoek onder 64 bezitters, waarvan 15% aangaf een 128 te willen kopen. Dat alleen al betekent ongeveer een half miljoen machines. Ook de voorverkopen, en ook in Nederland zijn bv. MCN, Computerworld en Cafka al actief met dergelijke aanbiedingen, loopt erg goed en beloopt al een paar honderdduizend machines. Positief nieuws, en dat was ook wel nodig na een toch bepaald slecht jaar voor Commodore. Niet te zien op de CES, maar wel nagenoeg gereed is de Amiga, die met superbe ikoon/muis graphics, met 256 KB RAM, 3 1/2 inch microfloppies, hele geavanceerde geluids en video-chips en gebaseerd op de 32 bits Motorola 68000 chip familie, de microwereld nog zal verbazen. Volgens veel deskundigen wordt dit de Apple-killer bij uitstek.



Marshall Smith

Nut

De belangrijkste vraag, de ondertoon die de hele hobbycomputermarkt bezighoudt, was in Chicago natuurlijk of

men er in zal slagen nieuwe zinvolle toepassingen voor de huiscomputer te brengen. De grote groep gebruikers blijkt het opstapje naar de serieuze toepassingen als boekhouding en tekstverwerking niet te maken. Ze hebben wel een tekstpakket en database, maar echt actief aan het schrijven (op)slaan is gewoon niet populair bij de meerderheid van de kopers. Dan blijft dus programmeren, recreatieve (spel) en educatieve software over.
Langzamerhand weten de hardwareleveranciers ook wel, dat zij aan het brengen van nieuwe toepassingen niet zoveel meer kunnen doen. Ze kunnen meer geheugen inbouwen, betere graphics en een muis toevoegen, maar de software moet het doen. Ze kijken dan ook angstvallig naar de softwarehuizen voor nieuwe, interessante programma's. En die komen er ook wel. Met de print/graphics programma's als Printshop en Newsroom is een hele nieuwe koers ingeslagen en met de zelfmaak-boeken zoals de Playwriter serie van Woodbury kunnen kinderen iets tastbaars maken. Dat is de grote trend van 1985, software die "**TANGIBLE** is, die tastbare resulaten oplevert.

Software

In Amerika zijn op dit moment de diverse Flight Simulators, Karateka en Bruce Lee vechtspelen, een aantal adventures zoals Zork, Print Shop, Music Shop en Dazzle Draw van Broderbund en natuurlijk Ghostbusters het meest populair. Een paar nieuwe programma's, die ons opvielen bij onze rondgang over de show waren nieuwe actiespelen van Epyx, waarbij natuurlijk WINTER GAMES en de EIDOLON en KORONI's RIFT, en een nieuwe adventure van Infocom, met de titel WISHBRINGER. Nogal wat film/software combinaties werden getoond, zoals A View to a Kill naar de nieuwste James Bond, een programma dat overigens al bij MCN in ons land te koop is (ƒ 49,-). Steven Spielberg's nieuwste film THE GOONIES heeft een computerneefje, terwijl maker Datasoft daarvan met ALTERNATE REALITY ook een zeer kunstige beeldadventure uitbracht, waar u meer van zult horen. Animatie ofwel het maken van bewegende figuren kan met Broderbunds FANTAVISION (op de Apple II), maar we vonden zelf voor kinderen Activision's FIREWORKS, om zelf een vuurwerk te maken, nog leuker. Ook THE LUESHER PROFILE van Mindscape werkt met bewegende objecten. Van Mindscape is ook RACTER, een soort computertaalkunstprogramma, waarbij een dialoog met de computer in de ELIZA traditie wordt opgebouwd en tot een soort literatuur kan leiden.

The Fourth Protocol

Literatuur per computer, het klinkt als een paradox, maar de ontwikkeling van de "Adventures" gaat in snel tempo die richting uit. De Britse topschrijver Frederick Forsyth heeft (met hulp van de uitgever Hutchinsons) zijn laatste bestseller "The Fourth Protocol" omgetoverd in een even spannend en zeer vernuftig computerspel. The Fourth Protocol is een spannende internationale spionagethriller, vol met geheime agenten, bommen, MI5, Nato en veel actie. Het computerspel is helemaal gelijk aan het boek, het bestaat uit drie opeenvolgende delen, die alleen maar toegankelijk zin met behulp van een verborgen codewoord uit het voorgaande deel. Vooral het eerste deel, een avontuur met nu eens geen tekst, maar ikoonbesturing, is buitengewoon origineel en razend knap uitgevoerd. Wanneer je alle drie delen achter de rug hebt, blijft de laatste en moeilijkste opdracht over: het onschadelijk maken van een atoombom binnen dertig seconden. De sleutel voor het de-activeren van de bom zit in een onbeduidend lijkend stukje informatie ergens eerder in het spel.
Gelukkig is dit Fourth Protocol spel niet helemaal identiek aan de plot van het boek, het zou anders veel te makkelijk zijn voor de miljoenen lezers die het boek inmiddels verslonden hebben. Het spel is in de U.K. met groot enthousiasme ontvangen, binnen de kortste keren waren er 50.000 van verkocht. Omdat men ook op de wat oudere computergebruiker mikt komt er na de versie voor de C-64 en de Spectrum waarschijnlijk binnenkort ook een versie voor de IBM PC en de Apple II op de markt.
In Engeland kost The Fourth Protocol bij 13 pond, en het is voorlopig dus alleen interessant voor wie de engelse taal goed machtig is.

Sideways

Voor wie last heeft met uitgebreide databases met veel velden, grote spreadsheets (Rekenmatrices) of boekhoudingen, die niet gewoon op zijn printer passen, is er nu een oplossing. Met Sideways kan men namelijk de hele zaak dwars uitprinten. De beperking van 40, 80 of 132 kolommen in de breedte is daarmee opgeheven. Bij computers als de Apple en de PC's is dit pakket al erg populair, we gebruiken het zelf voor veel dingen en op de 64 en 128 zal het zeker zijn nut bewijzen. Zeker door de keuze van lettertypes (men kan namelijk kiezen veel groottes en soorten letters) kan men hiermee met zijn matrixprinter (want dat die blijft is onontbeerlijk) de fraaiste resultaten ontfutselen. Het pakket komt van Timeworks (USA 312-9489200) en kost in de VS 30 dollar. Een van de pakketten, waarmee Sideways zonder problemen werkt, is **Swiftcalc** van Sideways.

Gebruikersgroepen

VCGN (070-971851) zit iedere laatste zaterdag van de maand in het Lodewijk Makeblijde lyceum in Rijswijk. Inlichtingen over de vele afdelingen:
Zeeland 01180-25746 West-Brabant 01640-51639 Midden-Brabant 01620-55634 Oost-Brabant 04120-42054 Zuid-Limburg 043-476935 Zuid-Holland 010-501023

PET Benelux heeft bijeenkomsten in Zeist (1e zaterdag),Nijmegen (2e zaterdag) en Haaksbergen (3e zaterdag). (Inl.05759-2211).

HCC Compet gebruikersgr. 03404-59599.

HOT NEWS uit Amsterdam in de Meervaart op de 2e zaterdag van de maand en ook in Zwolle (038-548459), Midden Nederland (03453- 1375) en Deventer (05700-50758).

WZW uit Belgie, met afdelingen in Antwerpen, Limburg, West- Vlaanderen inl. Bindstr. 19, te Berchem Antwerpen.

In Arnhem zit ook een Commodore-club, waar men clubavonden op dinsdag iedere twee weken organiseert en ook cursussen geeft. Inl. Talingsingel 32 te Velp, tel. 085-647782.

MISSERS

Syntax checksum.
Helaas zijn er bij de plaatsing van dit handige programma in C-info 5 van deze jaargang een paar storende foutjes geslopen. De checksum van de checksum kwam niet overeen met het programma.

De gepubliceerde Basic listing is korrekt. De checksumgetallen wijken af. Regel 95 bestaat niet in deze versie, dus die kan vergeten worden. Verder luiden de checksumgetallen voor regels 240, 250 en 260 als volgt: 114,89,68. Sorry voor dit ongemak. Als je de juistheid van dit bericht zelf kunt testen, ben je er zeker van dat jouw checksumprogramma werkt.

VIC Circuit
In regel 425 op pagina 26 van nr. 5 ontbrak de verwijzing achter GOTO, het moest GOTO 600 zijn.

In het artikel over het zelf aanpassen van een TV gingen we ervan uit, dat een aantal delen van de TV geaard zouden zijn. Dat is soms niet het geval, er kan een fase van de wisselstroom op dat chassis staan en dan is de hele operatie wat gevaarlijker. Meet vooraf eerst even uit, of er nergens spanning op staat.

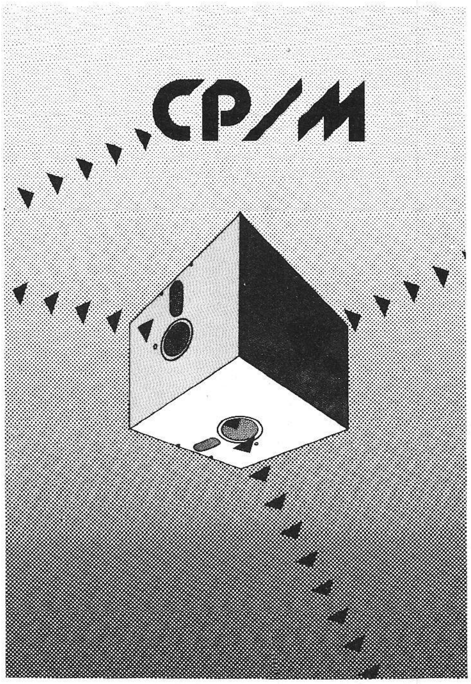
nabestellen

ABONNEMENTEN
OUDE NUMMERS

Abonnementen kosten *f* 55,– en voor een vol jaar Commodore-Info (10 à 11 nummers) is het verreweg het gemakkelijkste, wanneer u dat bedrag aan ons overmaakt op giro 1585491 tnv. SAC Blaricum of op onze bank in België BBL nr. 310050602562

OUDE NUMMERS
Reeds verschenen nummers zijn na te bestellen. Maar, helaas, NIET per brief, kaart of telefonisch. We leveren die oude nummers alleen bij vooruitbetaling op onze giro 1585491. Jammer voor degenen, die ons al vroegen om toezending, maar helaas is dat administratief niet te verwezenlijken, extra girokaarten maken en sturen zou echt te veel kosten. De prijs is *f* 6,75 per nummer (dus aangeven welk nummer).

De belofte van toegang tot het grote reservoir van CP/M software is verleidelijk. Diskdrive-incompatibiliteit maakt het allemaal wat minder aantrekkelijk maar er is een methode om CP/M programma's toch in de 64 te krijgen. W. Maessen (080-787031) geeft een programma om CP/M-programma's te „importeren".



DEEL 2

CP/M OP DE COMMODORE-64

CP/M: Control Program for Microcomputers

In deel 1 van deze serie, no. 2, jrg. 2, gingen we globaal in op het hoe en waarom van CP/M op de COMMODORE-64. In dit deel wordt een programma gegeven, waarmee het mogelijk is, CP/M programma's van andere CP/M-computers binnen te halen.

Commodore zelf lijkt de steun voor de CP/M module op de 64 te hebben opgegeven, hij wordt tenminste niet meer geleverd, zegt men. Maar goed, wie er een heeft of te pakken kan krijgen heeft hier met de komende CP/M golf voor de 128 vast iets aan. We gaan ook in op een aantal programma's die voor CP/M beschikbaar zijn en (indien bekend) bekijken we of het importeren hiervan veel problemen oplevert.

Programmatuur voor CP/M-64

Zoals eerder opgemerkt, is er voor CP/M-64 nog geen software op de markt. Dit klinkt enigszins tegenstrijdig ten opzichte van het verhaal over de grote standaardisatie die CP/M met zich mee zou brengen. Wat is het geval?

Alle CP/M-software zou inderdaad rechtstreeks op de CBM-64 kunnen draaien, ware het niet dat de manier waarop CP/M-64 zijn disks format, totaal afwijkt van de manier waarop andere CP/M-machines dit klusje klaren. Een diskette van bijvoorbeeld Apple(-CP/M) is voor de CBM-64 een raadsel: Bad Track or Sector. Hetzelfde geldt voor alle andere CP/M-machines. En Commodore zelf zal misschien, heel misschien, met een schijfje komen. Wanneer, kunnen ze niet zeggen. Blijft eigenlijk nog 1 manier over: CP/M-programma's binnenhalen via de gebruikerspoort vanuit een andere machine.

Hoe?

Zoals al eerder opgemerkt, is het mogelijk de user-port aan te spreken mits men eigen driver- routines schrijft. De eenvoudigste manier is om de 6510 het grootste gedeelte van de kommunikatie te laten doen: via een 6510-assemblerprogramma wordt het hele beschikbare geheugen (plm. 35 K) volgeschreven met dat wat via de userport binnenkomt, we schakelen over naar de Z80, deze leest het geheugen uit en zet het op disk. Dit is veruit de eenvoudigste manier: het vergt geen speciaal programma op de zendende machine want PIP ptp: = prog. naam

GEHEUGENINDELING COMMODORE 64 ONDER CP/M Z80

Z-80 adres	6510 adres	Funktie (bovengrens)
$EFFF	$FFFF	6510 kernal
$E000	$F000	6510 I/O
$D000	$E000	Vrije ram of CP/M 48K
$C000	$D000	BIOS80
$AFFF	$BFFF	BDOS
$AA00	$BB00	CCP
$9C06	$AC06	TPA
$9400	$A400	Zero page Z80
$0100	$1100	
$0000	$1000	Eigen 6510 programma, BIOS call 8
$FF00	$0F00	Eigen 6510 programma, BIOS call 7
$FE00	$0E00	BIOS65
$FA00	$0A00	Data uitwisseling Z80- 6510
$F900	$0900	Disk I/O buffer
$F800	$0800	Zero page 6510
$F000	$0000	

voldoet. Nadeel is alleen, dat eventuele transmissiefouten niet opgespoord kunnen worden. Beter is dan om met de zendende machine een protokol af te spreken, bijvoorbeeld: zend 128 bytes van het programma, bereken een 1 byte checksum en stuur deze over, de 64 berekent ook de checksum en stuurt een ACK of NACK naargelang Ehij dezelfde waarde van de checksum vindt als welke doorgestuurd is, de 64 schakelt de Z80 in om naar schijf te schrijven indien het goed was of hij ontvangt dezelfde 128 bytes opnieuw. Enzovoorts. De bijgaande programma's gebruiken de eerste methode. Programma's volgens de tweede methode zijn in ontwikkeling maar vereisen het bezit van een tweede (CP/M-) computer.

Het binnenhalen van CP/M-programma's

De hierbij afgedrukte programma's, de een in 6510-assembler, de ander in 8080-kode, zijn samen in staat CP/M-programma's tot een maximale lengte van plm. 35 K in de Commodore-64 te laden. Er wordt geen protokol gehandhaaft om eventuele transmissiefouten op te sporen. Daartegenover staat het voordeel dat er ook Egeen programma voor de zendende komputer geschreven hoeft te worden. Wat betreft de hardware: voor de CBM-64 is alleen een RS- 232 interface benodigd om de kommunikatie met de zendende komputer tot stand te kunnen brengen. Dit interface is voor plm. f. 100,- bij V&D verkrijgbaar. Het is ook mogelijk om het interface zelf te bouwen met behulp van de twee standaard RS- 232 driver chips MC1488 en MC1489 en een 7404 inverter. Voor het installeren van de programma's gaan we als volgt te werk. Begin met het Z80 programma (programma 1). Voer het in via ED, assembleer en load. De uiteindelijke COM-file die daarvan het resultaat is moet daarna aangevuld worden met het 6510-programma. Voor het implementeren van het 6510-programma is een machinetaal-monitor zoals Micromon een eerste vereiste. Via deze monitor kan programma 2 (in 6510- assembler) vanaf adres 0E00 hex in het geheugen gezet worden. Is dit karwei geklaard, dan verplaatsen we de routine naar het RAM-gebied vanaf C000 tot C200 hex. Dit

```
                 ; ****************************
                 ;        F I L T R A N
                 ; ****************************
                 ;  W. Maessen  1984
                 ;  Program to download programs
                 ;  from other CP/M-machines
                 ;  into the CBM-64 in CP/M-mode
                 ; ****************************
                 ;
0100 =           tpa       equ      0100h    ; start address program
0005 =           bdos      equ      0005h
0080 =           dmaddr    equ      0080h
005c =           fcb       equ      005ch
0000 =           boot      equ      0000h
000d =           cr        equ      0dh
000a =           lf        equ      0ah
                 ;
0100                       org      tpa
                 ;
0100 312f05                lxi      sp,stack
0103 c3a604                jmp      start
                 ;
                 ; screen handling routines
                 ;
0106 0e09        print:    mvi      c,9
0108 cd0500                call     bdos
010b c9                    ret
                 ;
010c 0e02        outchr:   mvi      c,2
010e cd0500                call     bdos
0111 c9                    ret
                 ;
0112 0e01        getchr:   mvi      c,1
0114 cd0500                call     bdos
0117 c9                    ret
                 ;
0118 e5          adprt:    push     h
0119 cd0601               call     print
011c e1                    pop      h
011d 7c                    mov      a,h      ; digit 1
011e e6f0                  ani      0f0h     ; strip ls nybble
0120 0f                    rrc               ; rotate right 4 times
0121 0f                    rrc
0122 0f                    rrc
0123 0f                    rrc
0124 cd3e01                call     conv
0127 7c                    mov      a,h      ; digit 2
0128 e60f                  ani      0fh
012a cd3e01                call     conv
012d 7d                    mov      a,l      ; digit 3
012e e6f0                  ani      0f0h
0130 0f                    rrc
0131 0f                    rrc
0132 0f                    rrc
0133 0f                    rrc
0134 cd3e01                call     conv
0137 7d                    mov      a,l      ; digit 4
0138 e60f                  ani      0fh
013a cd3e01                call     conv
013d c9                    ret
                 ;
                 ; convert decimal to ascii and print
                 ;
013e e5          conv:     push     h
013f fe0a                  cpi      0ah      ; digit?
0141 da4901                jc       dgt      ; yes, jump
0144 c637                  adi      37h
0146 c34b01                jmp      p
0149 c630        dgt:      adi      '0'
014b 5f          p:        mov      e,a
014c cd0c01                call     outchr
014f e1                    pop      h
0150 c9                    ret
```

```
                ;
                ; clear keybd-buffer
                ;
0151 3e00       kbdclr:  mvi     a,0
0153 32c6f0              sta     0f0c6h
0156 c9                  ret
                ;
                ; check file-name
                ;
0157 116403     filchk:  lxi     d,noname ; fill with noname-message
015a 3a5d00              lda     fcb+1
015d fe20                cpi     ' '      ; is first byte of name space?
015f ca0805              jz      done     ; yes, stop program
0162 118503              lxi     d,badnam ; search for ? in name
0165 215c00              lxi     h,fcb
0168 3e3f                mvi     a,'?'
016a 0610                mvi     b,16
                ;
016c be         qloop:   cmp     m
016d ca0805              jz      done     ; yes, stop program
0170 23                  inx     h
0171 05                  dcr     b
0172 c26c01              jnz     qloop
0175 c9                  ret              ; filename ok, continue
                ;
0176 118000     filsav:  lxi     d,dmaddr ; set output buffer for disk
0179 cd3402              call    setdma
017c 11e002              lxi     d,filsv
017f cd0601              call    print
                ;
                ; open file
                ;
0182 115c00              lxi     d,fcb    ; delete old file
0185 cd3a02              call    delete
0188 115c00              lxi     d,fcb    ; open new file
018b cd4002              call    make
018e 11a703              lxi     d,nodir
0191 3c                  inr     a
0192 ca0805              jz      done
0195 118000              lxi     d,dmaddr
0198 cd3402              call    setdma
                ;
019b 115003              lxi     d,opnok
019e cd0601              call    print
                ;
                ; compute # of 128-byte blocks
                ; reg. de contains top of buffer
                ; reg. hl contains bottom address
                ;
01a1 2aac02              lhld    bottom
01a4 110d03              lxi     d,stad
01a7 cd1801              call    adprt
01aa 2a05f0              lhld    0f005h
01ad 0600                mvi     b,0      ; add $62 (y-reg index)
01af 3a62f0              lda     0f062h
01b2 4f                  mov     c,a
01b3 09                  dad     b
01b4 7c                  mov     a,h      ; correct for 6510
01b5 de10                sbi     10h
01b7 67                  mov     h,a
01b8 11fa02              lxi     d,endad
01bb cd1801              call    adprt    ; print end address
01be 23                  inx     h        ; set eof mark
01bf 3e1a                mvi     a,1ah
01c1 77                  mov     m,a
01c2 2b                  dcx     h
01c3 7c                  mov     a,h
01c4 fe92                cpi     92h      ; over $9200 = $a200 6510 ?
01c6 dacf01              jc      nxt3     ; no, ok
01c9 22a802              shld    top      ; keep old top address
01cc 210092              lxi     h,9200h  ; set temp. top address
01cf eb         nxt3:    xchg
01d0 2aac02              lhld    bottom   ; load start address
01d3 7b         nxt1:    mov     a,e      ; divide by 128 shifting
```

RAM-gebied wordt, indien CP/M in de 44 K-versie loopt, niet door CP/M beïnvloed. Nadat het programma verplaatst is (vergeet niet het eerst op disk te saven), wordt CP/M geladen. Laad via DDT de COM-file in het geheugen van de Commodore. Indien alles goed is gegaan met het invoeren van het Z80-programma moet het 6510 gedeelte naar Z80-adres 0531 hex getransporteerd worden. Dit kan met het DDT kommando: MB000, B200, 531 (Move from B000 to B200 to position 0531). Geef een control-C (Stop-toets) en geef na het booten het kommando SAVE 8 FILTRN.COM.

Gebruik

Het programma is nu klaar en kan als volgt gebruikt worden.

❶ Geef onder CP/M het kommando 'FILTRN file.ext', waarbij Efile.ext de naam is waaronder het te ontvangen programma of bestand op disk gesaved moet worden

❷ Op het scherm verschijnt: 110 OR 300 BAUD (1/3) ? Geef 1 als de overdracht met 110 BAUD moet plaatsvinden, 3 indien dit met 300 BAUD dient te gaan. Het gebruik van hogere baudrates heeft in verband met de traagheid van de 64 geen zin, hij kan het niet volgen.

❸ Het programma komt dan in een interaktieve modus. Er wordt een terminal gesimuleerd. Dit is handig indien een file overgehaald wordt van een groot multi-user main-frame of mini. Er kan dan vanuit de 64 een kommando gegeven worden om de file, die overgehaald moet worden te LISTEN o.i.d. Geef geen RETURN, dat doet het file-transfer programma zelf.

❹ Indien we zover zijn dat we de file gaan ontvangen drukken we de '←'-toets in. Op het scherm verschijnt: STARTING FILETRANSFER - . Vanaf dit moment moeten de data van de andere machine binnen gaan komen. Wordt gewerkt met een andere mikro, dan kan daar het kommando: PIP PTP:= file.ext gegeven worden. Dit kommando stuurt de data uit de file naar de kommunikatie- poort. Deze

```
01d4 95              sub     l          ; the difference begin-end
01d5 4f              mov     c,a
01d6 7a              mov     a,d
01d7 9c              sbb     h
01d8 47              mov     b,a
01d9 a7              ana     a          ; clear carry
01da 79              mov     a,c        ; 16-bits shift
01db 17              ral                ; multiply by 2
01dc 78              mov     a,b        ; load acc with hi-byte of
                                          difference
01dd 17              ral                ; shift left
01de c601            adi     1          ; add one to round
01e0 4f              mov     c,a        ; c contains # of 128-byte
                                          records
01e1 6f              mov     l,a
01e2 2600            mvi     h,0
01e4 c5              push    b
01e5 112003          lxi     d,blks
01e8 cd1801          call    adprt      ; print # of blks used
01eb c1              pop     b
01ec 2aac02          lhld    bottom
01ef 118000  agn1:   lxi     d,dmaddr; start block moving
01f2 0680            mvi     b,128
01f4 7e      agn2:   mov     a,m        ; start loop 128
01f5 12              stax    d
01f6 23              inx     h
01f7 13              inx     d
01f8 05              dcr     b
01f9 c2f401          jnz     agn2       ; last byte of block ?
01fc c5              push    b
01fd d5              push    d
01fe e5              push    h
01ff cd4602          call    write      ; write to disk
0202 11cb03          lxi     d,dfull    ; check if disk full
0205 b7              ora     a          ; then a<>0
0206 c40805          cnz     done
0209 e1              pop     h
020a d1              pop     d
020b c1              pop     b
020c 0d              dcr     c
020d c2ef01          jnz     agn1
0210 3aa902          lda     top+1
0213 fe00            cpi     0
0215 ca2702          jz      endsav     ; last byte read
0218 2aa802          lhld    top        ; go save $b000-$c000
021b eb              xchg
021c 2100b0          lxi     h,0b000h; reset start-pointer
021f 3e00            mvi     a,0        ; reset hi-byte top
0221 32a902          sta     top+1
0224 c3d301          jmp     nxt1
0227 115c00  endsav: lxi     d,fcb
022a cd4f02          call    close
022d 11dd03          lxi     d,memsav
0230 cd0601          call    print
0233 c9              ret
             ;
             ; bdos calls
             ;
0234 0e1a    setdma: mvi     c,26
0236 cd0500          call    bdos
0239 c9              ret
             ;
023a 0e13    delete: mvi     c,19
023c cd0500          call    bdos
023f c9              ret
             ;
0240 0e16    make:   mvi     c,22
0242 cd0500          call    bdos
0245 c9              ret
             ;
0246 115c00  write:  lxi     d,fcb
0249 0e15            mvi     c,21
024b cd0500          call    bdos
024e c9              ret
```

data worden in het 64-geheugen opgeslagen en tegelijkertijd op het scherm geechood. Indien men ziet dat het eind van de file bereikt is drukt men op de '↑'-toets. Het programma geeft op het scherm de begin- en eind adressen van het programma in het geheugen en begint de file naar schijf te schrijven.

❺ Indien dit gebeurd is komt het programma weer in de interaktieve modus. De gebruiker krijgt dan de Ekans op een eventueel main-frame uit te loggen. Met de STOP- toets kan het programma dan beëindigd worden. Het programma is in staat 8-bits kodes (dus ook machinetaal) te ontvangen, mits op de zendende machine de kommunikatie ook op 8-bits gedefinieërd is. Verdere instellingen van de kommunikatie zijn: 110 of 300 BAUD, 3-line, no-parity, full duplex, 2 stop bits.

Software

Het aantal programma's dat voor CP/M op de markt is, is fenomenaal. Voor de meest uiteenlopende toepassingen zijn wel programma's geschreven en daarom is het ook ondoenlijk ze hier allemaal te beschrijven.

In de loop van de tijd zijn er echter een aantal van deze programma's standaard geworden. Een paar hiervan heb ik via de boven beschreven methode in de Commodore binnengehaald. En wie schetst mijn verbazing, toen de meeste elk van deze programma's, en dat zonder aanpassingen, meteen bleken te werken. Welke waren dat?

MBASIC

In de eerste plaats MBASIC. Indien vergeleken met Commodore-Basic (ook Microsoft) is dit een gigant. Heeft CBM-Basic een 70-tal kommando's of keywords, bij MBASIC zijn dat er meer dan 130! Hierbij vinden we dan allerhande mogelijkheden van gestruktureerde statements zoals IF.... THEN....ELSE en WHILE...WEND, edit funkties zoals AUTO, RENUM en DELETE, disk- funkties (NAME oldname AS newname, FILES (directory), CHAIN (om Basic-programma's aan elkaar te koppelen) en kommando's om fouten op te sporen, c.q. op te

```
                   ;
024f 0e10      close:   mvi      c,16
0251 cd0500             call     bdos
0254 c9                 ret
                   ;
                   ; communication related calls
                   ;
0255 111d04    comm:    lxi      d,stcom
0258 cd0601             call     print
025b cd5101             call     kbdclr
025e 3e08               mvi      a,8
0260 3200f9             sta      0f900h
0263 3e01               mvi      a,1
0265 3200ce             sta      0ce00h
0268 00                 nop
0269 cd5101             call     kbdclr
026c c9                 ret
                   ;
026d 3e07      filet:   mvi      a,7
026f 3200f9             sta      0f900h
0272 cd5101             call     kbdclr
0275 3e01               mvi      a,1
0277 3200ce             sta      0ce00h
027a 00                 nop
027b cd5101             call     kbdclr
027e c9                 ret
                   ;
027f 113303    bdset:   lxi      d,baud
0282 cd0601             call     print
0285 cdf201             call     getchr
0288 fe31               cpi      '1'
028a ca9d02             jz       b1
028d fe33               cpi      '3'
028f c27f02             jnz      bdset
0292 3e80               mvi      a,080h
0294 32a0ff             sta      0ffa0h
0297 3e06               mvi      a,06h
0299 32a1ff             sta      0ffa1h
029c c9                 ret
029d 3e80      b1:      mvi      a,080h
029f 32a0ff             sta      0ffa0h
02a2 3e83               mvi      a,083h
02a4 32a1ff             sta      0ffa1h
02a7 c9                 ret
                   ;
                   ;
                   ; ***** start of program ****
                   ; temp storage area
                   ;
02a8 00000000  top    dw 0,0
02ac 00000000  bottom dw 0,0
                   ;
                   ;      **** text ****
                   ;
02b0 0d0a535441stfile db cr,lf,'starting filetransfer ->',cr,lf
02cc 555345205e       db 'use ↑-key to end',cr,lf,lf,'$'
02e0 0d0a535441filsv  db cr,lf,'start saving file....',cr,lf,'$'
02fa 0d0a454e44endad  db cr,lf,'end address   : $'
030d 0d0a535441stad   db cr,lf,'start address : $'
0320 0d0a232031blks   db cr,lf,'# 128-blks    : $'
0333 0d0a313130baud   db cr,lf,'110 or 300 baud (1/3) ? : $'
0350 0d0a5b2046opnok  db cr,lf,'[ file opened ]',cr,lf,'$'
0364 0d0a3f2046noname db cr,lf,'? filename must be specified',cr,lf,'$'
0385 0d0a3f2041badnam db cr,lf,'? ambiguous files not allowed',cr,lf,'$'
03a7 0d0a5b204enodir  db cr,lf,'[ nodirectory space available ]',cr,lf,'$'
03cb 0d0a5b2044dfull  db cr,lf,'[ disk full ]',cr,lf,'$'
03dd 0d0a5b2046memsav db cr,lf,'[ file saved ]',cr,lf,'$'
03f0 0c0d0a2046hdr    db 0ch,cr,lf,' file transfer program'
0409 0d0a202042       db cr,lf,'  by w. maessen',cr,lf,'$'
041d 0c0d0a2053stcom  db 0ch,cr,lf,' start of communications now:',cr,lf
043f 555345205f       db 'use ←-key to start filetransfer',cr,lf
0460 52554e2f53       db 'run/stop key to cancel program ',cr,lf,lf,'$'
0483 0d0a0a5b20endprg db cr,lf,lf,'[ e n d o f p r o g r a m ]',cr,lf,'$'
                   ;
```

vangen (TRON, TROFF, ON ERROR GOTO).

MBASIC doet hierdoor veel professioneler aan dan Commodore's eigen BASIC-versies en nogmaals: het werkte meteen!

FORTRAN

Dat laatste geldt ook voor het gehele pakket programma's dat behoort bij FORTRAN-80 (F80). Met deze FORTRAN komen ook een aantal programma's binnen bereik, die de Commodore onder CP/M gelijke mogelijkheden geven als wat men van grote, professionele computers kent. Dat is M80, een makro-assembler die in staat is zogenaamde 'relocatable files' te produceren, stukjes programmakode waarvan de absolute adressen nog niet ingevuld zijn, waardoor het mogelijk is hele subroutine-bibliotheken op te bouwen. Via LIB80, een ander programma in het pakket, is het mogelijk deze subroutine-bibliotheken te onderhouden. L80 is een 'linker', die hoofdprogramma's en subroutines aan elkaar hangt en daarvan 1 programma maakt. Samen met FORTRAN biedt dit pakket bijna onbegrensde faciliteiten om programma's te maken, deels bestaande uit een hogere programmeertaal, deels uit een zeer krachtige assembler.

Wordstar

Een ander typisch CP/M programma geeft meer problemen: Wordstar. Dit tekstverwerkingspakket is bedoeld voor een 80-koloms scherm en dat heeft mijn 64 nu eenmaal niet. Een 80-tekenskaart kan hier uitkomst brengen. Bovendien schijnt het mogelijk te zijn Wordstar zo te wijzigen, dat hij ook 40 koloms aankan. Hoe, dat is me nog niet bekend.

Compilers

Naast het bovengenoemde FORTRAN, bestaan er ook een groot aantal andere compilers onder CP/M. Dat zijn onder andere PASCAL, PL/1, BASCOM (Basic-compiler), COBOL en, niet te vergeten, de tegenwoordig erg populaire taal C. Deze laatste compiler is zelfs in C geschreven!

De meeste van deze compilers genereren code, die relocatable is en die daardoor makkelijk aan te vullen is met machine-kode subroutines.

```
04a6 11f003     start:    lxi      d,hdr
04a9 cd0601               call     print
                ;
                ; check given filename
                ;
04ac cd5701               call     filchk
                ;
                ; move 6510-code to $0e00-$0fff
                ;
04af 213105               lxi      h,beg65
04b2 1100fe               lxi      d,0fe00h
04b5 7e         nxt2:     mov      a,m
04b6 12                   stax     d
04b7 23                   inx      h
04b8 13                   inx      d
04b9 7a                   mov      a,d
04ba fe00                 cpi      0
04bc c2b504               jnz      nxt2
                ;
04bf cd7f02               call     bdset
04c2 213105               lxi      h,beg65
04c5 7d                   mov      a,l
04c6 fe80                 cpi      80h
04c8 dad404               jc       lt80      ; lo-byte < 80h
04cb 2e00                 mvi      l,0
04cd 7c                   mov      a,h
04ce c601                 adi      1
04d0 67                   mov      h,a
04d1 c3d604               jmp      stset
04d4 2e80       lt80:     mvi      l,80h
04d6 22ac02     stset:    shld     bottom    ; keep address
04d9 7c                   mov      a,h
04da c610                 adi      10h       ; correct for 6510
04dc 323cfe               sta      0fe3ch    ;hi-byte start
04df 7d                   mov      a,l
04e0 3238fe               sta      0fe38h    ;lo-byte
04e3 cd5502               call     comm
                ;
04e6 3afffe     wkloop:   lda      0feffh
04e9 fe00                 cpi      0
04eb ca0505               jz       stopit
                ;
                ; perform filetransfer
                ;
04ee 11b002               lxi      d,stfile
04f1 cd0601               call     print
04f4 cd6d02               call     filet
                ;
                ; End, save data
                ;
04f7 cd7601               call     filsav
                ;
                ; Possibility to logoff
                ;
04fa 3e00                 mvi      a,0
04fc 32fffe               sta      0feffh
04ff cd5502               call     comm
0502 c3e604               jmp      wkloop
                ;
0505 118304     stopit:            lxi      d,endprg
0508 cd0601     done:              call     print
                ; error exit
050b e1                   pop      h         ; remove return address
                                               from stack
050c c30000               jmp      boot
050f                      ds       32        ; local stack area
052f =          stack     equ      $
0531 =          beg65     equ      $+2
052f                      end
```

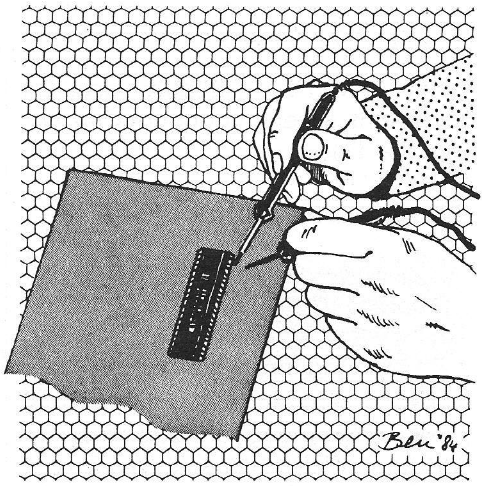
Programma 1: Z80 programma. Voer dit in via de ED-editor onder CP/M, assembleer en road.

```
.: 0e00 4c 30 0e ad a0 0f 8d 94
.: 0e08 02 ad a1 0f 8d 93 02 a9
.: 0e10 02 a0 02 a2 93 20 bd ff
.: 0e18 a9 80 a2 02 a0 ff 20 ba
.: 0e20 ff 20 c0 ff a9 a3 85 f8
.: 0e28 a9 a2 85 fa 60 ea ea ea
.: 0e30 20 03 0e a0 00 84 62 a9
.: 0e38 80 85 05 a9 15 85 06 a2
.: 0e40 80 20 c9 ff a9 0d 20 d2
.: 0e48 ff a2 80 20 c6 ff 20 e4
.: 0e50 ff 85 8b ad 97 02 85 8c
.: 0e58 20 b0 0f f0 35 a5 8c c9
.: 0e60 00 d0 e6 a2 03 86 9a a5
.: 0e68 8b ea a4 62 91 05 20 d2
.: 0e70 ff c8 84 62 c0 ff d0 d1
.: 0e78 e6 06 a0 00 84 62 a5 06
.: 0e80 c9 a1 d0 0a a9 c0 85 06
.: 0e88 d0 bf ea ea ea ea c9 d0
.: 0e90 d0 b7 a9 80 20 c3 ff a9
.: 0e98 00 85 c6 85 99 a9 03 85
.: 0ea0 9a a9 ff 85 91 60 ea ea
.: 0ea8 ea ea ea ea ea ea ea ea
.: 0eb0 8a c9 41 90 06 c9 5b b0
.: 0eb8 03 69 20 60 c9 61 90 06
.: 0ec0 c9 7c b0 03 e9 20 60 c9
.: 0ec8 7f 90 fb e9 5f 4c b1 0e
.: 0ed0 8a c9 41 90 06 c9 5b b0
.: 0ed8 03 69 80 60 c9 61 90 06
.: 0ee0 c9 7c b0 03 e9 1f 60 c9
.: 0ee8 7f 90 02 e9 5f 4c d1 0e
.: 0ef0 a9 ff 8d ff 0e 4c 92 0e
.: 0ef8 ea ea ea ea ea ea ea 00
.: 0f00 20 03 0e 20 e1 ff d0 03
.: 0f08 4c 92 0e a2 00 86 99 20
.: 0f10 e4 ff f0 16 85 6a c9 5f
.: 0f18 d0 03 4c f0 0e a2 80 20
.: 0f20 c9 ff a6 6a 20 b0 0e 20
.: 0f28 d2 ff a2 80 20 c6 ff 20
.: 0f30 e4 ff aa 20 d0 0e 85 6a
.: 0f38 a2 03 86 9a 20 55 0f a5
.: 0f40 6a f0 c0 a9 20 20 d2 ff
.: 0f48 a9 14 20 d2 ff a5 6a 20
.: 0f50 d2 ff 4c 03 0f a5 a2 c9
.: 0f58 15 10 01 60 a0 00 84 a2
.: 0f60 a4 6b f0 0c a0 00 84 6b
.: 0f68 a9 20 20 d2 ff 18 90 13
.: 0f70 a0 ff 84 6b a9 12 20 d2
.: 0f78 ff a9 20 20 d2 ff a9 92
.: 0f80 20 d2 ff a9 9d 20 d2 ff
.: 0f88 60 ea ea ea ea ea ea ea
.: 0f90 ea ea ea ea ea ea ea ea
.: 0f98 ea ea ea ea ea ea ea ea
.: 0fa0 80 06 ea ea ea ea ea ea
.: 0fa8 ea ea ea ea ea ea ea ea
.: 0fb0 a2 00 86 99 20 e4 ff 85
.: 0fb8 07 a2 03 86 9a 20 d2 ff
.: 0fc0 a5 07 c9 5e 60 ea ea ea
```

Programma 2: Voer in via MICROMON

Tot zover een beschrijving van de voor CP/M beschikbare programmatuur. CP/M wordt voor Commodore-liefhebbers steeds interessanter. De pas aangekondigde C-128 heeft namelijk ook een CP/M- modus waarbij de Z80-processor op 4 MHz draait. Dit in tegenstelling tot die op de CBM-64 die, omdat de VIC-chip niet sneller kan, het op een snelheid van 1 Megahertz moet doen. ●

Vragenrubriek.



Vragen staat vrij!

Door Jan Bodzinga

De vakantie is al bijna voorbij, de zon schijnt toch nog af en toe na een slecht begin van de zomer, en de stapel vragen over de Commodores wordt maar weinig lager. Voor ons dus weer veel werk, maar laat dat geen reden zijn om je vraag niet aan ons voor te leggen.

De spelregels zijn simpel, stuur je vragen op aan de redaktie, en zet in de linkerbovenhoek 'Vragenrubriek/ Jan Bodzinga'. Wij doen ons best om voor alle problemen een oplossing te verschaffen, als het niet in C-info is, dan wel langs een andere weg. Van problemen die inmiddels in z'n algemeenheid in ons blad zijn behandeld nemen we aan, dat die geen aparte oplossing meer nodig hebben.

C-16 geheugenuitbreiding.

John Hendriks uit Amsterdam vraagt of er inmiddels geheugenuitbreidingen te krijgen zijn, die kunnen worden aangesloten op de uitbreidingspoort van z'n C-16. Hij heeft daarover gelezen in 'Van start met de C-16' maar kan ze nergens bemachtigen.

Zoals inmiddels wel algemeen bekend zal zijn, heeft Commodore een scheve schaats gereden met het uitbrengen van de PLUS/4 en C-16 computers. Men had gerekend op een enorm succes, maar de apparaten zijn niet aangeslagen, met het gevolg, dat er een grote verliespost op het Commodore-conto verscheen, maar ook, dat de kopers van deze apparaten in feite in de kou kwamen te staan.
Want Commodore, noch andere bedrijven, zijn van zins om veel ontwikkelingen te doen voor een verliesgevende markt. We kregen laatst wel wat software en bv. Logo is er ook wel voor de Plus/4, maar juist niet (ons bericht daarover was te voorbarig) voor de C-16. Dat betekent dus in de praktijk, dat je weinig of niets zult kunnen krijgen voor de C-16 computers, maar een paar zaken hebben daar iets voor. Dit geldt dus ook voor de geheugenuitbreiding.
't Is spijtig genoeg, maar wij doen ons best en hebben ondertussen contact opgenomen met een paar Engelse leveranciers van geheugenuitbreidingen en zullen daarover berichten. Bij voldoende belangstelling kunnen we eventueel via Infolist wat importeren voor deze minderheidsgroep.

Checksum tips.

Er zijn nogal wat lezers, die problemen hebben met het programma SYNTAX CHECKSUM. Na intypen en kontrole blijkt het toch niet te werken. De heer Van De Heuvel uit Utrecht meldt daarover:

Urenlang heb ik zitten kontroleren maar de checksum bleef alle diensten weigeren. Totdat ik ontdekte, dat de checksum, zowel als de snellader die ik gebruik, worden aangeroepen met dezelfde SYS-opdracht. Dat bleek uiteindelijk de oplossing. Snellader weg, checksum erin, en het loopt als een trein.

Een goede tip, die niet alleen geldt voor het checksum programma maar van toepassing is op alle software die op dezelfde geheugenplaatsen opereert. Bij het debuggen van dit soort geheimzinnige fouten moet je altijd uitgaan van een kale computer, zonder alle aangehangen toeters en bellen als Toolkits, printerdrijvers en wat dan niet meer. Blijkt de fout dan nog op te treden, dan weet je zeker dat het aan de software ligt.

Tape gebruik.

P. Weedage uit Breda wil graag bepaalde listings die bedoeld zijn voor gebruik met disk veranderen, zodat hij er met een cassetterecorder mee kan werken.

In het algemeen is het mogelijk, en zelfs vrij simpel om een Basic programma om te zetten van disk naar tape gebruik of omgekeerd. Om dit te doen, moeten in de eerste plaats alle OPEN commando's uit zo'n programma worden opgezocht. Daarin moet het devicenummer worden aangepast. Dat is de tweede parameter uit dit commando. Staat er b.v. OPEN 2,8,3,"filenaam,seq,read", dan is dit een opdracht voor de diskdrive (device 8). Wil je dit aanpassen voor tape (device 1), dan moet het devicenummer een 1 worden. Verder is het bij cassette-gebruik van belang of er wordt gelezen of geschreven, en daarvoor wordt het secundary adres -S.A.- (in het voorbeeld 3) gebruikt. Bij diskgebruik heeft dit adres weinig betekenis. Voor lezen van tape is dit S.A een 0, bij schrijven wordt dit een 1 (soms een 2). Dit S.A. bepaalt dus de status van de recorder. Daarom kan de toevoeging achter de filenaam (seq,write) ook uit de listing worden verwijderd. Bij het omzetten van tape naar disk, moet deze juist worden toegevoegd. (seq,read / of seq,write).
Ons voorbeeld komt er dus zo uit te zien.:
OPEN 2,1,0, "filenaam"
Helaas hoef je over het aanpassen voor cassettegebruik niet te denken als er in het programma gebruik wordt

gemaakt van zg. RELative files. De tape kent alleen programma en sequential files.

Device nummers.

In aansluiting hierop een vraag van Piet Grootjans uit Groningen. Hij wil graag iets meer weten over de diverse apparaten en hun devicenummers.

In de vorige vraag is (hoop ik) al het nodige duidelijk gemaakt, maar voor de volledigheid hier een overzicht van wat randapparaten en hun devicenummers etc.
Tape lezen : dev = 1 SA = 0
Tape schrijven : dev = 1 SA = 1 (write file+EOF = end of file)
Tape schrijven : dev = 1 SA = 2 (idem + EOT = end of tape)
Toetsenbord : dev = 0 SA = onbelang
Scherm : dev = SA onbel.
Printer : dev = 4 SA vele (funktie afh.van printer)
Modem : dev = 2 SA diverse (zie Prog.guide)
Disk : dev = 8 SA = 0 / 1 Directory lezen / schrijven.
Disk : dev = 8 SA = 15 Foutkanaal.
Disk : dev = 8 SA 3 to 14 of 16 tot 128
Dit zijn de belangrijkste mogelijkheden van randapparaten en hun devicenummers. Aan de hand hiervan is het mogelijk om b.v. gegevens voor de disk bestemd door eenvoudigweg het devicenummer te veranderen op het scherm of de printer te laten verschijnen.

Saven van Machinetaal.

J.B.Kroon uit Breukelen zoekt naar een mogelijkheid om z'n eigen MT-programma's via snellaad weg te schrijven naar tape of disk.

Het wegschrijven van machinetaal is minder eenvoudig dan het saven vanuit Basic. De meeste snelladers zijn hierop niet aangepast, alhoewel er een turbotool bestaat die dit wel kan. De save en laad routine voor machinetaal is nl. een andere dan de normale routine, waardoor snelladers het laten afweten. Een mogelijkheid is de machinecode eerst om te zetten naar Basicdata (zie listing DATA-MAKER) of met hulp van een MT-monitor te relocеren naar het Basic-gedeelte van de computer.
Als laatste (en beste) kun je nog proberen om de adressen in de nul-pagina waar de start ($(2B)= 43/44) en het eind ($(2D)= 45/46) van een Basicprogramma worden opgeslagen, te veranderen in de begin en eind adressen van jouw stuk machinetaal, waarna je simpelweg de speedlader/saver z'n werk laat doen. De computer denkt dan dat er gewoon met een Basic-programma wordt gewerkt.

Printer besturingskodes.

N.J.Staal uit Coevorden heeft een Juki printer aangesloten op een Vic. Dat werkt prima, maar in de handleiding van de printer staan allerlei mogelijkheden die zijn aan te sturen met hulp van besturingskodes. Zijn vraag: Hoe doe ik dat.

Het gebruiken van de vele mogelijkheden die de moderne printers bieden is min of meer gestandaardiseerd (gelukkig), zodat na verloop van tijd de software niet meer printer afhankelijk hoeft te zijn. Om een printerfunktie aan te sturen, moet er een serie karakterkodes naar de printer worden gestuurd. Deze series bestaan uit 1 of meer karakters, die echter altijd beginnen met een escape-karakter. Daarna komt de rest van de serie, die de feitelijke funktie bepalen.
De printerhandboeken geven meestal een lijst van besturingskodes, waarbij -omwille van de algemeenheid- i.p.v. CHR$(x), symbolen worden gebruikt. Zo is code 7 b.v. omschreven als BEL en code 27 heet ESC (escape). Deze code moet je vertalen naar de ASCII-waarde van de Commodore-computers. Een tabel daarvan vind je in het handboek (p.135). De symbolen staan daar echter niet bij, vandaar een paar van de meest gebruikte :
0=NUL, 7=BEL, US=31, RS=30, ESC=27, GS=29, SO=14, SI=15.
In de meeste printerhandleidingen staat de volledige tabel afgedrukt. Wil je via Basic nu de printer activeren met de besturingskode 'ESC + US + 9' dan doe je dat met het commando:

```
OPEN 4,4 : PRINT 4,CHR$(27);
CHR$(31);CHR$(9) : CLO SE 4
```

Hopelijk kun je aan de hand van dit voorbeeld de rest zelf invullen.

Pokenummers.

D.J.v.d.Vaart uit Vlaardingen vraagt om publikatie van een aantal POKE nummers die geheimen voor hem herbergen. Hij begrijpt soms niet waarom er moet worden gepoked.

De C-64, en ook de VIC hebben een simpele Basic-interpreter, waardoor bij die computers via Basic van veel mogelijkheden geen gebruik kan worden gemaakt. Omdat dingen als Hires en sprites wel degelijk zijn ingebouwd is er in feite maar 1 manier, om ze te benaderen en dat is machinetaal.
Via Basic kan dit worden gesimuleerd door het gebruik van POKE instructies, die niet meer doen, dan een geheugenlokatie voorzien van een andere waarde.
Het gaat te ver om op deze plaats alle mogelijke adressen te gaan bespreken, -we hebben al eens een aantal handige pokes gepubliceerd-, maar

het zal nuttig zijn te weten, dat de computer adressen doorgaans worden opgegeven in hexadicimale getallen (een getal met een $ ervoor) terwijl Basic alleen decimale getallen accepteert.
Om dus een goed gebruik van POKE te kunnen maken, moet je de hexadecimalen omrekenen.

Interface Writenow.

G.H.Nieuwveen uit Wormer krijgt z'n ROM tekstverwerker niet gekoppeld met de Micro Plus printer, die opzichzelf uitstekend werkt met Basic programma's.

Alle software, die ontwikkeld is voor de C-64, heeft beperkingen op het gebied van de printerinterfacing. Het is nog altijd ondoenlijk om voor iedere printer een passende routine in de software te verwerken. Dat komt niet zozeer door de printers, als wel door de vreemde gedragingen van het team, dat de Commodore machines heeft ontwikkeld.
WriteNow is in principe geschikt voor meerder printers, maar niet voor meerdere interfaces. In dit geval kunt je alleen proberen of de kombinatie wel tot stand kan worden gebracht door een aander interface te gebruiken, waarbij in plaats van de gebruikerspoort de seriele (disk/printer) plug wordt gebruikt.

De C-16.

L.G.de Haas uit Delft zoekt serieuze software voor de C-16, o.a. Basicode 2, en heeft tevens problemen met de tekstverwerker uit 'Van Start met Basic 3.5'.

Zoals ik al vaker heb geschreven, voor de C-16 bestaat bijna geen software, laat staan serieuze software. Ook de ontwikkeling van toepassingen voor deze machine staat nagenoeg stil.
De toepassingen die door u worden gezocht zult u dus zelf moeten ontwikkelen, of misschien zijn er lezers, die voor u en ons nuttige programma's hebben gemaakt, en ze voor plaatsing toesturen ? Datzelfde geldt voor de Basicode.
In het programma Tekstverwerker uit bovengenoemd C-16 boek, ook uitgegeven door SAC en het enige Nederlandstalige boek over de C- 16 en PLUS/4, dat we kennen, zitten inderdaad wat storende foutjes en aangezien ik mede dat programma heb gemaakt, weet ik gelukkig ook welke. In regel 400 wordt met 'CSR TERUG' de cursortoets met de pijl naar links bedoeld. In 420 staat GOTO 450, dit moet worden GOTO 390. Regel 650 wordt : PRINT#
8,D$(I), regel 860 wordt : GET #
8,D$(TE): IF ST <> 0 THEN S1=1 en als laatste
875 : PRINT #4,D$(TE);:TE=TE+1.
Nadat deze (druk)fouten zijn verbeterd, loopt dit programma zoals het hoort.

VIC-software naar C-64.

H.v.d.Merwe uit Roosendaal is een beetje boos op ons, omdat we ook wel eens leuke programma's publiceren, die niet geschikt zijn voor de C-64, maar voor andere Commodore machines. Hij stelt voor om bij deze software te vermelden hoe het ook geschikt is te maken voor de C-64.

We kunnen deze vraag kennelijk niet vaak genoeg voor het voetlicht brengen, want er blijken altijd weer mensen te zijn, die hierdoor in de problemen raken. Het is nu eenmaal zo, dat de computers van Commodore tot nu toe geen van alle compatibel zijn. Als het aanpassen van een listing een simpele zaak is, dan kan in feite iedere gebruiker zondermeer de listing veranderen. Worden er echter veel PEEK, POKE en SYS opdrachten in verwerkt, dan is dit een bijna ondoenlijke zaak geworden. Als de heer v.d. Merwe zich hiervoor zou willen aanbieden, dan zijn wij de de eersten om z'n aanpassingen in C-info te plaatsen.

1541 Drive.

Peter Brand uit Oudleusden vraagt zich af, of het verstandig is een 1541 diskdrive aan te schaffen, want hij heeft van alle kanten nogal wat kritiek op dit apparaat gehoord.

Peter, als je werkelijk een diskdrive nodig hebt, dan is er maar een oplossing, de 1541. En die zijn de laatste tijd van een goede kwaliteit.
Het is inderdaad zo, dat de oudere modellen nogal uiteen liepen wat betreft DOS-versies en er traden ook de nodige hitteproblemen op. Maar, zover ik weet, behoort dat allemaal tot het verleden, al zijn ook de nieuwe 1541-drives aan slijtage onderhevig. In principe is het een redelijke aankoop.

Basic in EPROM.

R.Hoo uit Rotterdam bezit sinds kort een EPROM-programmer, maar weet eigenlijk niet hoe hij Machinetaal en Basic in een EPROM moet zetten, en daarna ook nog gebruiken.

Een EPROM-programmer is een nuttig apparaat, maar je moet wel weten wat je er wel en niet mee kunt bereiken. Basic-programmatuur in Eprom is een ondoenlijke zaak, omdat de C-64 er van uitgaat, dat Basic begint op een adres in RAM, en als je daar een EPROM neerzet, dan kun je vervolgens niets anders meer met de com-

puter doen, sterker nog, bij het opstarten krijg je al problemen.
Machinetaal in EPROM is een goede mogelijkheid, maar je moet dan wel een MT-relocator inbouwen, die ervoor zorgt, dat de MT naar de juiste geheugenadressen wordt weggezet, vanuit de EPROM, die in de expansiepoort zit. Ook dit is niet zo simpel te doen, zeker niet als je weinig weet van machinetaal. Ik adviseer je dan ook om je zo snel mogelijk te werpen op het programmeren in MT, dan kom je vanzelf op het punt, dat je de EPROM-programmer nuttig kunt gaan gebruiken.

Jan Bodzinga. ●

ERVARINGEN

"En Toen Deed Ie Het Niet"

Wie wel eens problemen heeft met een onwillige printer, heeft misschien iets aan onderstaande tip.

Je besluit op een gegeven moment om een printertje aan te schaffen, of krijgt er eentje voor je verjaardag. Zo'n eenvoudig 40-koloms apparaat. Vol goede moed sluit je hem aan op de 64 ... en dan .... doet ie het niet. Niet uit het veld geslagen ga je wat experimenteren. Je vraagt de STATUS op en dan blijkt al snel dat die zegt: 'time out' of 'device not present'.

Een dergelijk probleem kent u misschien ook, je zit dan meteen in de zoekbooom. Stapje voor stapje foutzoeken dus maar.
Je ontkoppelt (eerst de 64 uitzetten) je cassette recorder en wonder boven wonder: hij doet het. Nu zit je pas echt in de problemen, omdat je je cassette recorder nodig hebt om je programma's te laden en omdat je na het laden niet ongestraft de stekker van je recorder eruit kan trekken. Goede raad is duur. Je belt met je dealer, met Commodore support en anderen. Niemand weet je te zeggen wat te doen, behalve: het zal wel aan de 64 liggen. Stuur hem maar op voor reparatie. Nou het lag niet aan de 64. Bovenstaand probleem heb ik als volgt opgelost.

Oplossing

Open de cassette-recorder (4 schroefjes aan de onderkant losdraaien met een mini kruiskopschroevendraaier). Je ziet dan een kabelboom. Op het punt waar de kabelboom zich vertakt in een veelvoud van snoertjes zoek je de write-line. (Deze is verbonden met pin E-5 - zie handboek.) Indien je de stekker van de recorder even openmaakt kun je zien welke kleur dat draadje heeft. Dit snoertje knip je zo door dat je aan beide kanten ruimte genoeg hebt om beide einden te verlengen. Nadat je beide einden hebt verlengd moet je tussen de andere einden van de verlengsnoertjes een mini-schakelaartje solderen. De beste plaats voor dit schakelaartje is ca. 1 cm. onder het woordje "save" aan de bovenkant van je recorder. Hij zit dan niets in de weg. Sluit hem zo aan dat de cassette-recorder "aan" is als het hefboompje van de schakelaar naar je toe wijst, dus naar de voorzijde van de recorder. Snoertjes netjes isoleren, bedrading van de snoertjes zo leiden dat deze niets in de weg kunnen zitten, recorder dichtschroeven en klaar is Kees.
Zodra je wilt laden, dan zorg je dat de schakelaar "aan" staat en wil je printen dan kun je ongestraft de recorder uitschakelen met de schakelaar. Succes!

P.Cremer, Utrecht ●

128 Software

Het is nog wat vroeg om nu al uitgebreid over de 128 software te gaan schrijven, maar toch noemen we al een aantal pakketten om aan te geven, dat er aan gewerkt wordt. Behalve de 64 software, die zonder problemen op de 128 draait, komt er ook speciale software voor de zgn. 128 mode met 80 kolommen op het beeldscherm, dus ideaal voor zakelijke toepassingen.

Zo is er de "PERFECT" serie met een aantal zakelijke pakketten en het integrale pakket JANE 2 van Arktronics met Janewrite, Janelist en Janecalc. Commodore zelf komt met Micro-Illustrator, Typing Professor en "A Comprehensive course in Basic".
Van Timeworks (444 Lake Cook Road, Deerfield Ill. 60015 USA) kregen we ook een interessant trio door, dat per stuk 70 dollar gaat kosten en onderling data-compatibel is:

Wordwriter 128

Een tekstprogramma met ingebouwde spellingschecker van 85000 woorden en voor snel rekenwerk een calculatorfunctie. Alle gebruikelijke functies van een modern tekstpakket zijn aanwezig.

Data manager 128

Opslag en oproepen van informatie met dit database-pakket is uiterst eenvoudig. Ook het maken van rapporten en uitdraaien van bestanden, afdrukken van stickers, het gaat met functies als X- sort, X-search en X-chart erg eenvoudig.

Swiftcalc 128

Een rekenmatrix voor huis- en klein zakelijk gebruik.

Van **Sierra** is er het spel GATO voor de 128. **Batteries Included** heeft een 128 versie van THE CONSULTANT. ●

De koppeling van de 64 aan een HiFi geluidsinstallatie zou de muzikale kwaliteiten daarvan natuurlijk niet vergroten, maar wel beter hoorbaar maken. Mark Stoutjesdijk maakte er een koppeling voor en volgens hem hebben zowel computer als versterker het nog allemaal overleefd. Iets voor de kabelsoldeerders!

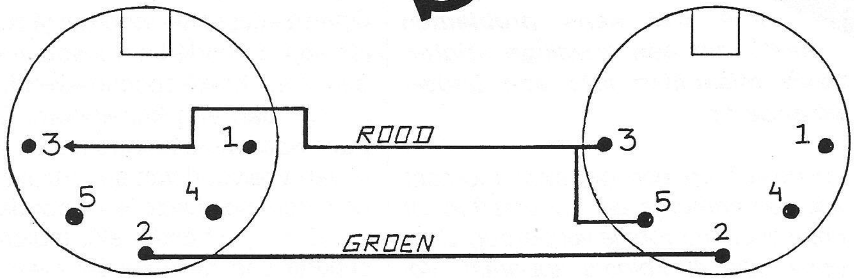
DE C-64 IN STEREO STEREO

Er wordt wel eens gezegd dat de 64 meer kan dan de gemiddelde synthesizer, maar dat is beslist onjuist. Tenminste, wanneer die synthesizer gekoppeld is met een heel systeem, dat bijvoorbeeld bestaat uit een mengpaneel, een electrische gitaar, een versterker, een equalizer, en tenslotte een electrische drum. Met dat mengpaneel zijn stereogeluiden te creëren en dat is iets, wat de C-64 met de beste wil van de wereld niet kan. Maar hij kan het wel benaderen, en met wat hulpmiddelen kan de mono-64 in ieder geval geluid uit de twee stereo kanalen laten komen.

Hier volgt een beschrijving van de aanpak om de C-64 met de geluidsinstallatie te verbinden en zo echt goed geluid te krijgen. Noch wel geen echte stereo, maar een benadering.

Het vervelende van de standaard geluidsinstallatie van de 64 is dat het geluid maar uit ééen luidspreker komt, namelijk die van de tv of monitor. Daar komt bij dat de kwaliteit daarvan vaak dermate slecht is, dat het geluid nauwelijks muziek mag worden genoemd, al speelt de Commodore Chopin of The Beatles. Daar moet wat aan te doen zijn, en wel op de volgende manier.

Versterkeringang



Audio/Video plug op C-64
2. aarde
3. audio out

2. aarde
3. linker kanaal
5. rechter kanaal

De audio-poort

De oplossing is eigenlijk super eenvoudig: de 64 heeft een separate audio/video uitgang (naast de seriële poort), en daar is wel wat mee te doen. Wie de User-Guide erop naslaat, vindt dat er 5 pennen zijn: de audio-in, audio-out, video-out, gnd, en de luminance. Van deze vijf zijn er maar twee te gebruiken voor wat we willen, nl. gnd (ground, aarde) en audio-out.

We willen nu met het signaal op die pennen de Commodore op een versterker aansluiten (AUX) en dan via de versterker het geluid van twee kanten zien te krijgen. Dus of door één box en de luidspreker van de monitor, of door twee boxen, wat een beter resultaat geeft. Daarvoor zijn een aantal connectors nodig. Afhankelijk van de gebruikte connectors op de versterker zijn er nodig:

Hebt u alles bij elkaar? Dan kan de verbinding worden gemaakt. Doe alles precies volgens de tekening, die geeft de standaard aansluitingen weer. Maar denkt u dat de versterker afwijkt van de standaard, kijk het dan liever even na in de handleiding of documentatie.

2 vijf pens DIN-pluggen (nr. 41/524)
ongeveer twee meter 2-aderige kabel
een versterker
een of twee luidsprekerboxen.

De kabelkleuren zijn groen en rood, groen als aarde en bij een kabel met een afschermende mantel dus liefst groen daaraan bevestigen. De kabel hoeft niet verschrikkelijk duur te zijn, het gaat om tamelijk gewone signalen. In de plug die in de versterkeraansluiting moet, moeten 3 en 5 doorverbon-

den worden, en aan de buitenkant moet een merkteken komen, dat duidelijk maakt of de plug in de 64 of de versterker thuis hoort.
Het is ook mogelijk, alleen de pennen 2 en 3 aan te sluiten, dus 2-2 en 3-3.

Tenminste, als de versterker onderscheid maakt tussen mono en stereo. En nog een vereiste is dan, dat de mono-input van de versterker pin 3 is, terwijl het vaak ook 5 is. Als de combinatie 2-2, 3-3 gemaakt kan worden, is geen merkteken nodig, immers de aansluitingen zijn aan iedere plug eender. Wat er nu dus gebeurt, is dat het geluid van de 64 op simpele wijze tegelijk aan twee ingangen wordt toegevoerd. In wezen zou die splitsing ook veel eleganter kunnen worden opgelost en met een electronisch filter het audiosignaal bv. gescheiden kunnen worden naar een aantal frequentiegebieden. Die moeten apart versterkt worden en dan zijn semi-stereo effecten te bereiken. In de praktijk zult u met deze eenvoudige splitsing echter heel wel uit de voeten kunnen.

Door wat met de balans te goochelen, kan het volume per box ingesteld worden en een redelijk resultaat worden bereikt, al is het dan geen stereo. En nog een belangrijk voordeel: de kwaliteit gaat er enorm op voorruit en u kunt de volumeknop nu voluit opendraaien. Bach uit de 64 begint op Bach te lijken en het wordt de moeite waard de computermuziek eens op de musicassette op te nemen! ●

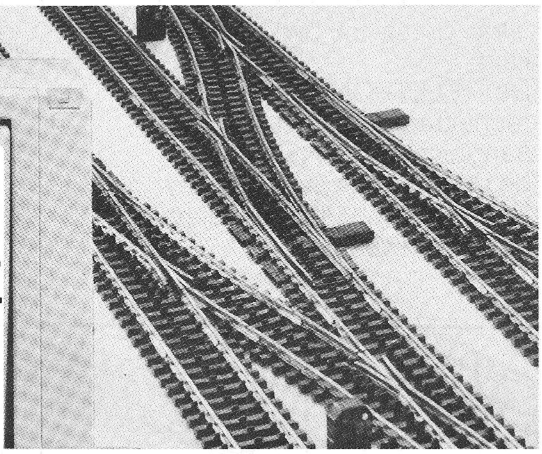
Tenslotte nog een lijstje met de geschatte kosten:

pluggen	*f* 1,75 per stuk
kabel	*f* 2,80 per meter
versterker	tussen 90,- en 500,- afh. van kwaliteit
boxen	tussen 50,- en 500,- afh. van kwaliteit

Maar meestal heeft u de boxen en de versterker al staan.

Knutselen met treintjes, electromotoren, relais en schakelaars, dat was voor velen in het pre-computertijdperk meer dan een hobby, ze waren er dag en nacht mee bezig.

ELECTRO DYNAMIEK



Op de Keulse computerbeurs waren er twee bedrijven druk bezig, die vergeten groep weer terug te brengen bij het grote (computerpeloton). Dat was Fischer Technik, ook in ons land al bezig met de verkoop van Computer-elektrobouwdozen en Märklin, een zeer bekende naam voor treintjes liefhebbers.

Fischer

Fischer Technik brengt bouwdozen voor het construeren van bewegende bouwsels. FT is na het helaas weggglijden in vergetelheid van MECCANO nu het toonaangevend constructiesysteem op de markt. Meegaand met de computertrend heeft men nu ook een koppeling met bv. de C-64 in het pakket. Via een interface kan men dan zelf (via relais) de besturing van motoren en dergelijke regelen en zelfs opnemen in gecompliceerde programma's om bv. robots te laten werken. Men heeft speciale Robot-bouwdozen, maar ook een plotter kan men met dit systeem bouwen. Via de interface (Voor de C-64 is dat Nr 30562, maar er is er ook een voor de 40XX en 80XX serie CBM micro's) worden via de user-poort de commando's doorgegeven naar de "real time" apparatuur.

De computer bestuurt de beweging

Per interface zijn er 4 uitgangen om motoren, lampen, ventilatoren, elektromagneten en derg. te bekrachtigen. Daarnaast 8 digitale ingangen om schakelaars, relais en sensoren op aan te sluiten, terwijl er ook nog twee analoge ingangen zijn om bv. potmeters of fotoweerstanden op aan te sluiten. Het besturen van de uitgangen kan met eenvoudige Basic opdrachten, waarvoor een uitvoerige handleiding beschikbaar is.

Märklin

Een andere invalshoek tot de modelbouw komt van Mäklin, traditioneel het kwaliteitsmerk voor HO treintjes. Onder de aanduiding **DIGITAL HO** heeft men treintjes en emplacementregeling nu ook toegankelijk gemaakt voor de computer als hoofdmachinist via een RS 232 interface. Met een eigen Märklin chip, die in de locomotieven wordt gemonteerd, word de zaak bestuurd. ●